ВЕСТНИК

ЮЖНО-УРАЛЬСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА

2023
Т. 12, № 1

ISSN 2305-9052

СЕРИЯ

«ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА И ИНФОРМАТИКА»

Решением ВАК включен в Перечень научных изданий, в которых должны быть опубликованы результаты диссертаций на соискание ученых степеней кандидата и доктора наук

Учредитель — Федеральное государственное автономное образовательное учреждение высшего образования «Южно-Уральский государственный университет (национальный исследовательский университет)»

Тематика журнала:

- Вычислительная математика и численные методы
- Математическое программирование
- Распознавание образов
- Вычислительные методы линейной алгебры
- Решение обратных и некорректно поставленных задач
- Доказательные вычисления
- Численное решение дифференциальных и интегральных уравнений
- Исследование операций
- Теория игр
- Теория аппроксимации
- Информатика
- Искусственный интеллект и машинное обучение
- Системное программирование
- Перспективные многопроцессорные архитектуры
- Облачные вычисления
- Технология программирования
- Машинная графика
- Интернет-технологии
- Системы электронного обучения
- Технологии обработки баз данных и знаний
- Интеллектуальный анализ данных

BULLETIN
OF THE SOUTH URAL STATE UNIVERSITY
2023
vol. 12, no. 1

SERIES
"COMPUTATIONAL MATHEMATICS AND SOFTWARE ENGINEERING"

ISSN 2305-9052

Vestnik Yuzhno-Ural'skogo Gosudarstvennogo Universiteta.
Seriya "Vychislitel'naya Matematika i Informatika"

South Ural State University

The scope of the journal:

- Numerical analysis and methods
- Mathematical optimization
- Pattern recognition
- Numerical methods of linear algebra
- Reverse and ill-posed problems solution
- Computer-assisted proofs
- Numerical solutions of differential and integral equations
- Operations research
- Game theory
- Approximation theory
- Computer science
- Artificial intelligence and machine learning
- System software
- Advanced multiprocessor architectures
- Cloud computing
- Software engineering
- Computer graphics
- Internet technologies
- E-learning
- Database processing
- Data mining

Содержание

Contents

УДК 004.932, 51-76, 57.087 **DOI: 10.14529/cmse230101**

МОДЕЛЬ ПРОГНОЗИРОВАНИЯ ЖИВОГО ВЕСА С ПОМОЩЬЮ ГЛУБОКОЙ РЕГРЕССИИ RGB-D ИЗОБРАЖЕНИЙ

© 2023 А.Н. Ручай[1,2,3]

[1] *Южно-Уральский государственный университет*
(454080 Челябинск, пр. им. В.И. Ленина, д. 76),
[2] *Челябинский государственный университет,*
(454001 Челябинск, ул. Бр. Кашириных, д. 129),
[3] *Федеральный научный центр биологических систем и агротехнологий РАН*
(460000 Оренбург, ул. 9 Января, д. 29)
E-mail: ran@csu.ru

Поступила в редакцию: 13.02.2023

Прогнозирование живого веса помогает контролировать здоровье животных, эффективно проводить генетическую селекцию и определять оптимальное время убоя. На крупных фермах для измерения живого веса используются точные и дорогостоящие промышленные весы. Взвешивание животного из-за стресса ведет к потере его веса и продуктивности на 5–10%. Однако, перспективной альтернативой является оценка живого веса с помощью морфометрических измерений животного, а затем применение уравнений регрессии, связывающих такие измерения с живым весом. Ручные измерения животных с помощью рулетки отнимают много времени и вызывают стресс у животных. Поэтому в настоящее время для бесконтактных морфометрических измерений все чаще используются технологии компьютерного зрения. В статье предлагается новая модель для прогнозирования живого веса на основе регрессии изображений с использованием методов глубокого обучения. Для регрессии изображений использовались RGB изображения и карты глубины вид сбоку для прогнозирования живого веса крупного рогатого скота. Показано, что на реальных наборах данных предложенная модель достигает точности измерения веса с ошибкой MAE 35.5 и MAPE 8.4 на тестовом наборе данных.

Ключевые слова: регрессия изображений, прогнозирование живого веса, глубокое обучение.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Ручай А.Н. Модель прогнозирования живого веса с помощью глубокой регрессии RGB-D изображений // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2023. Т. 12, № 1. С. 5–27. DOI: 10.14529/cmse230101.

Введение

В настоящее время существует два основных подхода к измерению веса тела [1]: использование промышленных весов и косвенные методы, основанные на взаимосвязи между морфологическими размерами тела и весом тела. Ручное измерение размеров тела животных занимает много времени, оно трудоемко и дорого. Отметим, что взвешивание животного из-за стресса ведет к потере его веса и продуктивности на 5–10%. Кроме того, это стресс как для работника, так и для животного. В настоящее время была разработана бесконтактная оценка морфометрических размеров с помощью недорогих датчиков и методов машинного зрения [2, 3]. Использование бесконтактной технологии существенно сокращает возможные временные затраты на проведение ручной и субъективной бонитировки для предсказания живого веса скота, исключает необходимые контактные измерения линейных промеров или прямого взвешивания скота с помощью весов.

Измерения размеров тела обычно используются для прогнозирования живого веса животных [4–6]. При этом для точного прогнозирования живого веса можно использовать измерения размеров тела вместе с другими факторами, характеризующими животное: возраст, пол, оценка состояния тела, генотип, объем тела, площадь тела и т.д. В большинстве последних исследований для прогнозирования живого веса животных использовался множественный линейный регрессионный анализ. Однако, эти традиционные методы неадекватны для точного прогнозирования [7]. Недавно несколько исследователей успешно применили различные алгоритмы машинного обучения для прогнозирования живого веса с использованием морфологических показателей животных [7–11]. Данные методы направлены на определение веса тела по морфологическим показателям животных. Эти исследования показали потенциал алгоритмов машинного обучения в точном прогнозировании нелинейной связи между живым весом и морфологическими признаками животных [7]. Более того, прогнозирование живого веса может быть основано на автоматически измеренных морфологических признаках с помощью системы двухмерного зрения [11, 12] и системы трехмерного зрения [3, 13]. В работе [14] предложена система оценки веса тела молочной коровы с ошибкой 5.2% на основе трех линейных обмеров, выполненных с помощью 3D-камеры. Однако, общим недостатком таких систем является то, что точность оценки веса зависит от модели прогнозирования, качества измерения морфологических признаков, выбора входных переменных модели, а также не достаточности апробации в связи с маленькой выборкой.

Другим перспективным направлением разработки модели прогнозирования живого веса является модель, созданная на основе регрессии изображений. Регрессия изображений является широко используемой задачей в компьютерном зрении для предсказания возраста, позы головы и ключевых точек лица [15]. Для прогнозирования живого веса крупного рогатого скота проще всего использовать RGB изображения (цветные изображения в цветовой модели RGB) и карты глубины (изображение, в котором для каждого пикселя вместо цвета хранится расстояние от объекта до камеры) [16, 17], или сами облака точек (набор точек модели в трехмерной системе координат), или восстановленные плотные целые трехмерные модели. Также важным вопросом является выбор вида положения животного для получения изображений. Боковой вид животного дает больше информации, однако его получить технологически сложней из-за требования укрепления и очистки камер. Вид сверху более приемлем в реальных условия фермы, так как нет подобных ограничений.

Цель данной работы — разработать надежную модель прогнозирования живого веса на основе регрессии изображений с помощью методов глубокого обучения. Для регрессии изображений использовались RGB изображения и карты глубины вид сбоку для прогнозирования живого веса крупного рогатого скота. Использование вида сбоку имеет следующие основные преимущества: отсутствие необходимости синхронизации данных между несколькими датчиками, отсутствие необходимости выполнения сложных трудоемких процедур внешней калибровки камеры, отсутствие необходимости выполнять реконструкцию плотной трехмерной модели животного, возможность использовать только одну камеру, что позволит удешевить технологию бесконтактного измерения живого веса животного.

Для обучения нейронной сети требуется высокое качество изображений, RGB изображения имеют разрешение 1920×1080, а карта глубины 512×424, что может искажать или неправильно передавать характеристики объекта. Нас интересует фильтрация RGB изображения и карты глубины с датчика RGB-D для улучшения ее качества [18].

Входом для глубокой нейронной сети может быть двухмерные RGB изображения или карты глубины, однако в будущем предполагается исследовать глубокие нейронные сети с входом с облаком точек. При ограниченном количестве доступных изображений у нас мало вариативности в данных, что может привести к переобучению. Стоит отметить, что выборка для обучения нейронной сети достаточно мала, поэтому необходимо дополнить обучающие данные синтезированными и модифицированными изображениями. Существуют два способа аугментации данных: с помощью дополнения сырых двухмерных RGB изображений и карты глубины, или более сложный — с большей вариативностью и модификацией, близкой к реальности, с помощью проецирования облака точек, полученных из карты глубины, на плоскость двухмерного изображения с ортогональной проекцией, так называемые 2.5D карты глубины. Предварительно из облака точек удаляется фон (удаление сцены с общего кадра), выравнивание позы животного, и затем дополняются жесткие преобразования в виде трехмерных вращений, масштабирований и перемещений. Мы выполнили проекцию облака точек как цветную компоненту (цветная проекция), так и карту глубины (2.5D карты глубины).

Основной вклад данной работы заключается в следующем:

- Были предложены методы для предобработки RGB изображений и карты глубины и создании цветной RGB проекции и 2.5D карты глубины для прогнозирования живого веса на основе регрессии изображений с помощью методов глубокого обучения.
- Был предложен метод трехмерной аугментации цветной проекции и 2.5D карты глубины с помощью жестких преобразований в виде трехмерных вращений и перемещений, что позволяет нам увеличить ограниченный набор данных и повысить эффективность прогнозирования живого веса при наличии вариаций позы, положения и масштаба животного.
- Были показали результаты на реальных наборах данных, которые демонстрируют, что предложенная модель с MAPE 8.1%, использующая цветные проекции и 2.5D карты глубины может достичь уровня точности измерения веса, превышающей тот, который достигается при традиционном взвешивании.

Дальнейшее изложение статьи построено следующим образом. В разделе 1 описаны необходимые методы и алгоритмы предобработки RGB-D изображений. В разделе 2 рассмотрена модель прогнозирования живого веса. В разделе 3 приведены результаты экспериментов с использованием предложенной модели. В заключении содержатся основные полученные результаты.

1. Предобработка RGB-D изображений

1.1. База данных

Наши эксперименты проводятся на двух открытых наборах данных из работ [2, 19]. Первый созданный набор данных содержит RGB-D данные, вес и ручные измерения 154 голов крупного рогатого скота — коров породы герефорд. 154 головы герефордских коров содержались на частной ферме с концентрированной кормовой подкормкой в возрасте от 12 до 15 месяцев. Вес коров составлял от 243 до 605 килограммов (кг). Набор данных собран системой сбора RGB-D изображений, состоящей из трех камер Microsoft Kinect v2. Две RGB-D камеры расположены с правой стороны C_1 и левой стороны C_2 прохода животных на расстоянии около 2.0 м от животного. Были использованы RGB изображения и карты глубины с правой стороны C_1 и левой стороны C_2. На рис. 1 показаны RGB изображения и карты

глубины крупного рогатого скота, снятые двумя камерами Kinect. В наших экспериментах были использованы RGB изображения и карты глубины с правой стороны C_1 и левой стороны C_2 отдельно. Также можно рассматривать RGB изображения и карты глубины с левой и правой стороны вместе. Для этого нужно отразить левую сторону в правую. Полный набор данных состоит из 5220 RGB изображения и 5220 карты глубины с правой стороны C_1 и 4620 RGB изображений и 4620 карт глубины с и левой стороны C_2 для 154 коровы.

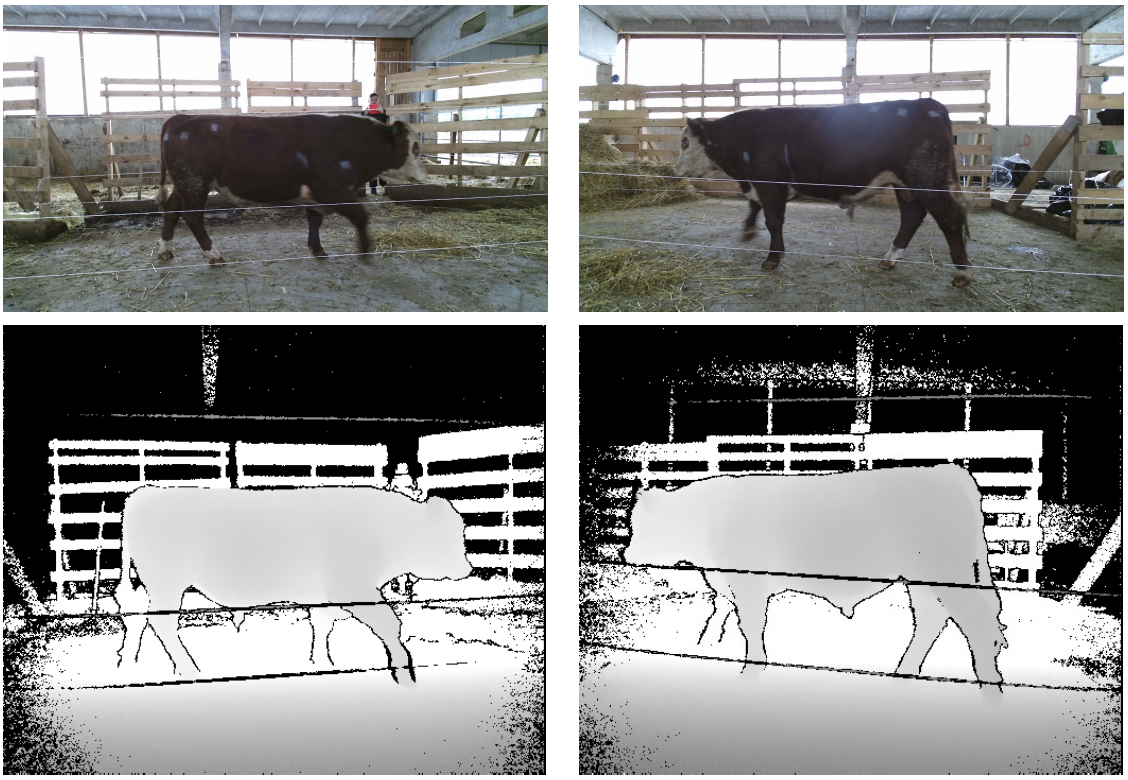


Рис. 1. RGB изображения и карты глубины коров были получены с помощью двух камер Kinect

Вторая использованная база включает в себя данные 121 особи молодняка крупного рогатого скота абердин-ангусской породы [19]. На момент экспериментов возраст животных составлял 16.5 месяцев, живой вес — 614.9 кг. Для каждого из 96 животных была собрана база данных, содержащая следующие данные: Номер RFID чипа, RGB изображения, карты глубин и облака точек, живой вес. Эта база данных находится в открытом доступе (https://github.com/ruchaya/CowDatabase2). Система захвата данных была установлена в проходе зала. Все измерения проводились на идущем животном с двух точек зрения, поскольку невозможно потребовать, чтобы животное остановилось и оставалось неподвижным. Две RGB-D камеры были расположены справа и слева от прохода животного на расстоянии около 2.0 м. В установке использовались две одинаковые камеры Microsoft Kinect v2, получающие изображения RGB и глубины с левой и правой стороны животного. Каждая камера глубины была подключена к ноутбуку, а все ноутбуки были подключены к локальной сети. Синхронно получаемые изображения в формате RGB-D записывались на соответствующий ноутбук для каждой камеры. Сбор и хранение данных были реализованы на основе Kinect v2 SDK. Каждая камера, инициализированная сигналом запуска,

начинала захват кадров с частотой 30 Гц. Время на ноутбуках было синхронизировано, и наилучшее соответствие облака точек могло быть выбрано в течение кратчайших временных интервалов между тремя устройствами. Разрешение RGB-изображений и изображений глубины составляет 1920 × 1080 и 512 × 424 пикселей, соответственно. На рис. 2 показаны RGB-изображения, карты глубины и облака точек крупного рогатого скота, снятые двумя камерами Kinect. Полный набор данных состоит из 4180 RGB изображения и 4180 карты глубины с правой стороны C_1 и 3860 RGB изображений и 3860 карт глубины с левой стороны C_2 для 121 коровы.

Рис. 2. RGB-изображения, карты глубины и облака точек крупного рогатого скота, снятые двумя камерами Kinect

1.2. Предобработка базы данных

Данные трехмерного сканирования коровы представляют собой динамические последовательности, содержащие пустые кадры без животного, целые кадры с полностью вошедшим животным и частичные, когда животное не вошло целиком. Однако, отсутствие части головы животного или задней части может легко привести к ошибкам прогнозирования живого веса. Задачу обнаружения кадров с целым животным можно свести к задаче детектирования частей животного. В работе [20] был предложен метод детектирования области головы, бедра и тела животного на двухмерном изображении RGB. Благодаря высокому разрешению RGB информации на изображении, результат обнаружения целевой области является более надежным, чем метод трехмерного детектирования. Была использована существующая модель обнаружения *YOLO v4* [21] для обнаружения нескольких областей разного размера на одном двухмерном изображении, и была дообучена модель для детектирования областей головы, бедра и тела животного на двухмерном изображении. Наличие трех областей — тело, бедро и голову животного — однозначно определяют кадр с целым

животным. Как показано на рис. 3, области идентифицируются тремя цветами, а детектированные области представлены двухмерным окном. Была обработана вся база данных для выделения только полных кадров с животным.

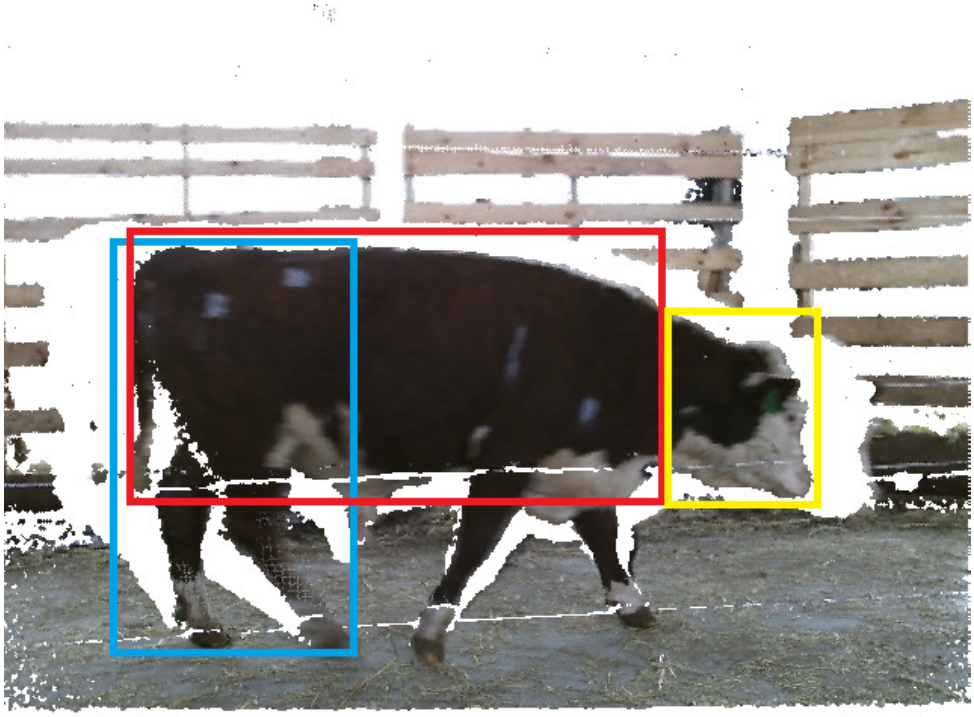


Рис. 3. Области тела, бедра и головы животного обозначены красным, синим и желтым цветом соответственно

Кроме того, на точность измерения и прогнозирования живого веса может влиять поза животного. Согласно [22], требования к правильной позе можно кратко сформулировать следующим образом: четыре копыта измеряемой животного должны составлять прямоугольник, а ветвь туловища должна быть почти прямой линией. Предложенный метод извлечения скелета [22] обеспечивает лучший способ оценки позы для последующего прогнозирования живого веса. Поскольку методы оценки живого веса могут быть восприимчивы к неправильным позам животного, то необходимо определить схему выбора позы, чтобы гарантировать, что выбрана правильная поза для последующих измерений. В последовательности данных сначала рассматриваем кадры, которые не помечены отсутствием ноги. Если нет ни одного немаркированного кадра, удовлетворяющего порогу, то используем те же критерии для выбора кадров с одной отсутствующей ногой. Если в последовательности нет удовлетворяющего порогу кадра, то кадр не выбирается в этой последовательности.

Первый обработанный набор данных состоит из 1701 RGB изображения и 1701 карты глубины с правой стороны C_1 и 1406 RGB изображений и 1406 карт глубины с левой стороны C_2 для 154 коров. Второй обработанный набор данных состоит из 1536 RGB изображения и 1536 карты глубины с правой стороны C_1 и 1327 RGB изображений и 1327 карт глубины с левой стороны C_2 для 121 коровы.

1.3. Шумоочистка RGB изображений

В работе был использован этап предварительной обработки для улучшения качества RGB изображения. В процессе записи или передачи информации на цветных цифровых изображениях часто возникает импульсный шум из-за неисправностей датчиков, ошибок передачи, аналого-цифровом преобразовании [23]. Чтобы улучшить качество цветных изображений, важно использовать эффективные подходы к оцениванию параметров искажения и последующему удалению импульсного шума [24, 25]. Был предложен новый подход к шумоподавлению цветного изображения с помощью морфологической фильтрации, где происходит обнаружение поврежденных пикселей и удаление обнаруженного шума посредством морфологической фильтрации. Предложенный алгоритм позволяет эффективно удалять импульсные помехи на цветных изображениях [26].

1.4. Шумоочистка карты глубины

Карта глубины описывается кусочно-гладкими областями, ограниченными резкими границами объекта, поэтому значение глубины меняется скачкообразно, и небольшая ошибка вокруг границы объекта может привести к значительным артефактам и искажениям. Кроме того, карта глубины зашумлена из-за отражения инфракрасного света, а отсутствующие пиксели без какого-либо значения глубины выглядят как черные дыры на картах глубины. Шум и дыры могут повлиять на точность прогнозирования живого веса, поэтому необходимо использовать алгоритмы шумоподавления и заполнения дыр.

Мы предлагаем использовать переключающийся двусторонний фильтр [27] для удаления шума с карты глубины, снятой с помощью камеры RGB-D. Переключение двусторонней фильтрации применяется не ко всем пикселям карты глубины, а только к тем, где могут быть шумы и дыры, то есть на границах и резких изменениях. Сначала обнаруживаются области с резкими изменениями и границами на изображении RGB, а затем фильтрация применяется только к соответствующим областям на карте глубины. Многочисленные эксперименты [27] показали, что переключающийся двусторонний фильтр дает лучшую производительность с точки зрения точности восстановления карты глубины и скорости среди распространенных алгоритмов удаления шума с карты глубины.

1.5. Шумоочистка облака точек

В последние годы было предложено множество методов фильтрации трехмерных облаков точек. Наши эксперименты показали [28], что алгоритм ROR, реализованный в PCL [29], дает лучший результат с точки зрения точности восстановления облака точек, рассчитанного с помощью расстояния Хаусдорфа, среди существующих алгоритмов. ROR удаляет выбросы, если количество соседей в определенном радиусе поиска меньше заданного порога. Мы можем указать количество соседей, которые должны быть в пределах заданного радиуса, чтобы оставаться в облаке точек. С помощью фильтра VoxelGrid с параметрами листьев (листья задают размер вокселя) для каждой оси на 0.01 уменьшается качество облака, путем аппроксимации их к центройду. Облако точек, подготовленное для проекции, обрабатывается фильтром RadiusOutlierRemoval, который удаляет точку, если количество ее соседей не равно 20 в радиусе 0.05. Пример обработанного облака точек показан на рис. 4.

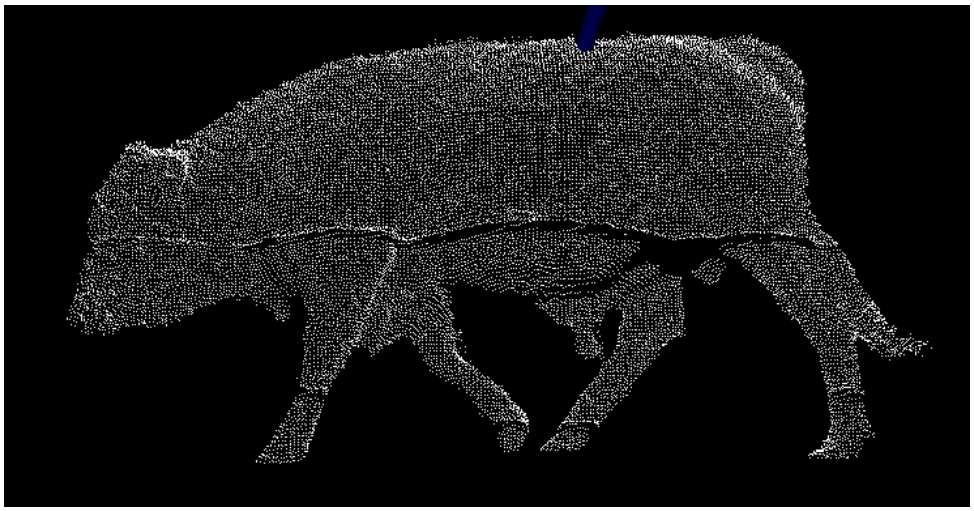


Рис. 4. Пример облака точек после предобработки

1.6. Удаление фона из облака точек

Важным шагом обработки облака точек является удаление фона из облака точек. Был разработан и реализован алгоритм для вычитания облаков точек. Сначала необходимо получить два облака точек: одно облако с пустым кадром без животного, другое облако с животным. Для этого мы использует всю последовательность кадров, полученную при записи, и также как для предобработки базы данных находим пустой кадр без животного. Далее каждая точка одного облака сравнивается с другим по компонентам (x, y, z координаты). Если абсолютная разница во всех компонентах превышает заданную дельту, то точка переносится в новое облако точек. Наилучший результат достигается при дельта равной 0.1. Пример работы алгоритма представлен на рис. 5: исходный кадр облака точек с животным, облако точек с фоном, и итоговое облако после вычитания облаков точек.

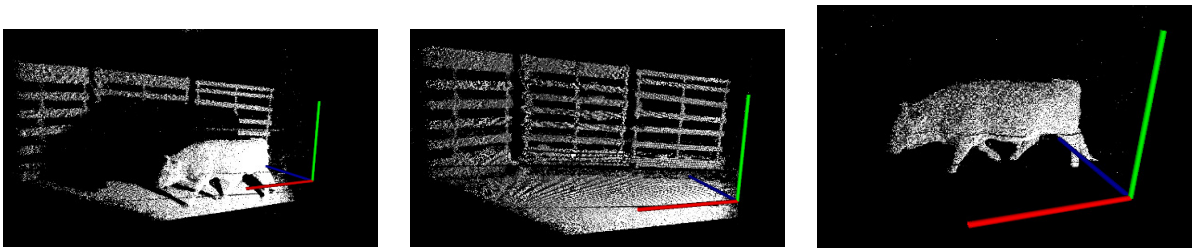


Рис. 5. Пример работы алгоритма: исходный кадр облака точек с животным, облако точек с фоном, и итоговое облако после вычитания облаков точек

1.7. Нормализация позы и вычисление линий симметрии

Двусторонняя симметрия является важным и универсальным понятием среди животных. Симметричная плоскость животного используется для получения осей X, Y, Z. Поиск подходящих ориентаций часто помогает в автоматизированном поиске и обработке трехмерных объектов. Кроме того, нормализация позы помогает алгоритмам машинного обучения учитывать информацию о позе, делая прогнозы распознавания объектов более точными.

Мы предложили быстрый алгоритм обнаружения двусторонней симметрии для облака точек [30, 31]. Сначала был использован алгоритм PCA для обнаружения начальной симметрии. Затем путем полного перебора плоскостей симметрии, проходящих через центр тяжести относительно исходной плоскости симметрии, определялась оптимальная плоскость симметрии с помощью модифицированной метрики Хаусдорфа.

Чтобы оценить точность и скорость предлагаемого алгоритма обнаружения симметрии на реальных, мы сравним предложенный алгоритм с алгоритмом PCA. Плоскости симметрии по осям X, Y, Z из алгоритма PCA отображаются в виде плоскости, закрашенной красным цветом, из предлагаемого алгоритма отображаются в виде плоскости, закрашенной зеленым цветом. Результаты обнаружения симметрии на реальных данных корова показано на рис. 6. Для отсканированной модели коровы плоскость симметрии по оси X из алгоритма PCA совпадает с предложенным алгоритмом. По осям Y и Z предложенный алгоритм в данном случае немного исправляет ситуацию.

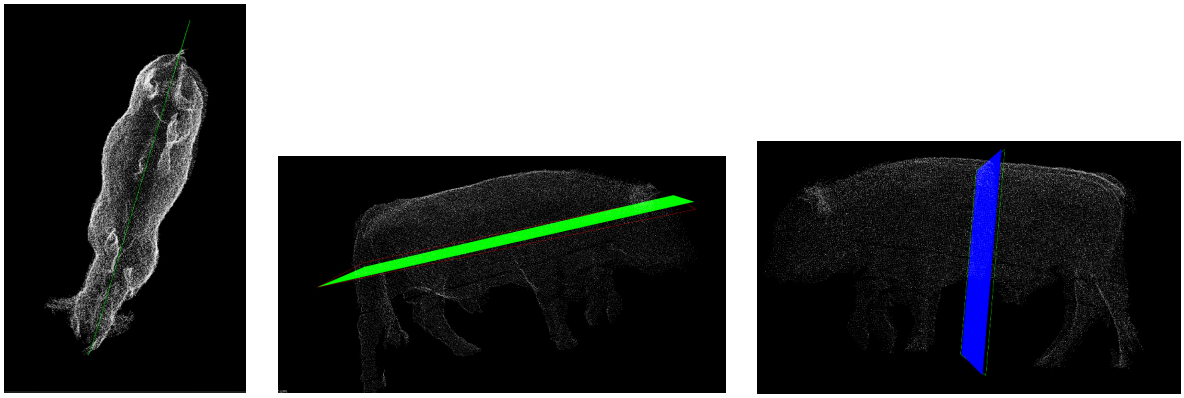


Рис. 6. Плоскости симметрии коровы

Для всего облака с помощью предложенного метода вычисляются линии симметрии. Далее, для выравнивания, облако с помощью найденных коэффициентов плоскости и декартова базиса собственного подпространства выравнивается поза животного параллельно нормированной плоскости OXZ. На рис. 7 приведен пример выровненного облако точек.

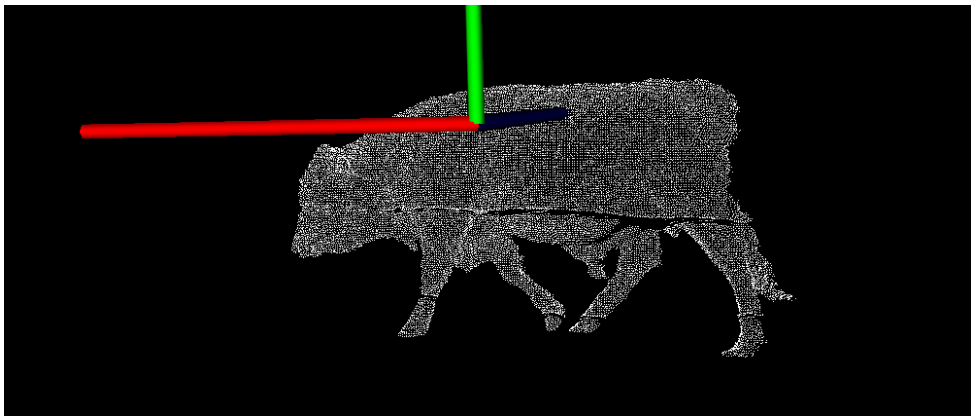


Рис. 7. Пример выровненного облако точек животного

1.8. Вычисление проекции карты глубины (2.5D карта глубины)

Был разработан и реализован алгоритм вычисления плоских проекций облаков точек. Полученное облако переносится к началу системы координат (т. е. крайняя точка параллелепипеда, в который вписано облако, перемещается в точку (0,0,0)). Средствами библиотеки OpenCV в памяти создается пустое изображение размером 299 × 150. Максимальными параметрами ширины и высоты животного выбраны значения 2.5 и 1.35 соответственно. Все координаты точек облака нормализуются к размеру изображения, и по координатам x и y точки из облака в пустом изображении устанавливается значение цвета z. Полученное изображение сохраняется в формате bmp. На рис. 8 показан пример такого выравнивания.

Рис. 8. Пример полученной проекции карты глубины (2.5D карта глубины)

1.9. Цветная проекция (RGB проекция)

Из-за того, что облака изначально являются сильно разряженными (количество точек в них относительно мало), их цветные проекций имеют большие расстояния между пикселами. Так как увеличение количества точек в облаке не приводит к приемлемому результату, то был разработан алгоритм для вычисления цветной проекции. Алгоритм основан на медианном фильтре в заданном окне. Задается радиус окна, центральный пиксел заменяется на среднее значение всех пикселов, попавших в окно, при этом условием замены является нахождение в квадрате не менее, чем k пикселов, потому как малое количество пикселов дает не приемлемый результат. В результаты многочисленных экспериментов, размер окна 7 дает приемлемый результат, в качестве k было выбрано значение 13. Сложность предложенного алгоритма вычисления цветной проекции оценивается как $O\left(\frac{nm}{s}\right)$, где n и m — ширина и высота изображения, s — площадь окна. На рис. 9 представлен пример цветной проекции.

1.10. Предобработка изображений для входа в нейронную сеть

Изменение размера (image resize). Разрешение карты глубины составляет 512 × 424 пикселей, а RGB изображения — 1920 × 1080 пикселей, поэтому необходимо было изменить размер изображения с учетом соотношения сторон, определяемого как $r = w/h$, где r — соотношение сторон, w и h — ширина и высота изображения, соответственно. Соотноше-

Рис. 9. Результаты вычисления цветной проекции при размере окна 5, 6, и 7, соответственно

ние сторон рассматривалось для изменения размера изображения как параметр, который помогает сохранить наилучшее качество исходного изображения в процедурах понижения и повышения размера изображения. Большинство предварительно обученных моделей используют размер изображения 224×224 (ширина, высота), который используется в качестве целевого размера. При повышении размера изображения использовалась кубическая интерполяция, а при понижении размера изображения наилучшие результаты дает интерполяция по площади.

Нормализация изображений (Feature standardisation). После предварительной обработки изображения карта глубины и RGB изображение имеют разные значения пикселей. Чтобы уменьшить это влияние на результаты предсказания, перед обучением модели данные необходимо нормализовать, чтобы обеспечить одинаковый порядок значений пикселей. Существует три основных метода масштабирования значений пикселей, поддерживаемых классом ImageDataGenerator из библиотеки Keras [32]: нормализация пикселей (масштабирование значений пикселей до диапазона [0,1]); центрирование пикселей (масштабирование значений пикселей до нулевого среднего значения); стандартизация пикселей (масштабирование значений пикселей до нулевого среднего значения и единичной дисперсии). Алгоритм стандартизации пикселей достигает наилучшей производительности. Алгоритм нормализации StandardScaler основан на удалении среднего значения и масштабировании до единичной дисперсии, и определяется как

$$v = \frac{x - u}{s}, \tag{1}$$

где x — текущее значение признака, n — нормализованное значение признака, u — среднее значение для всего обучающего набора данных, s — стандартное отклонение для всего обучающего набора данных.

2. Модель прогнозирования живого веса

Глубокое обучение — это общий метод машинного обучения, при котором модель обучается без специализированных алгоритмов под конкретные задачи, а использует иерархическое или многоуровневое обучение. Сверточная нейронная сеть (CNN), вероятно, является самой популярной архитектурой, используемой в настоящее время в компьютерном зрении. Глубокое обучение реализовано с помощью библиотеки Keras [32].

Модель MRGBDM (Model RGB and Depth Map) для прогнозирования живого веса коровы показан на рис. 10, на котором входными данными являются изображения RGB и карты глубины. Модель содержит 3 блока свертки (con1, conv2, conv3) и два полносвязанных слоя (FC и OUT). Блок conv1 имеет два слоя с 64 3×3 фильтрами, блок conv2 имеет два слоя с 128 3×3 фильтрами, и последний блок свертки имеет три слоя с 256 3×3 фильтрами. После блоков conv1, conv2 и conv3 используются слои подвыборки с размером ядра 2×2. Усеченное линейное преобразование (Rectified Linear Unit, ReLU), которое не показано на рис. 10, является функцией активации, применяемой после каждого сверточного слоя и полносвязного слоя.

Модель MRGB (Model RGB) для предсказания живого веса коровы показан на рис. 11, на котором входными данными являются только RGB изображения. Модель содержит 3 блока свертки (con1, conv2, conv3) и два полносвязанных слоя (FC и OUT). Блок conv1 имеет 32 3×3 фильтра, блок conv2 имеет 64 3×3 фильтра, и последний блок свертки имеет 128 3×3 фильтра. После блоков conv1, conv2 и conv3 используются слои подвыборки с размером ядра 2×2. Усеченное линейное преобразование ReLU, которое не показано на рис. 11, является функцией активации, применяемой после каждого сверточного слоя и полносвязного слоя.

Модель MDM (Model Depth Map) для предсказания живого веса коровы показан на рис. 12, на котором входными данными являются только карты глубины. Модель содержит 3 блока свертки (con1, conv2, conv3) и два полносвязанных слоя (full и out). Блок conv1 имеет 64 3×3 фильтра, блок conv2 имеет 64 3×3 фильтра, и последний блок свертки имеет 256 3×3 фильтра. После блоков conv1, conv2 и conv3 используются слои максимального объединения с размером ядра 2×2. Усеченное линейное преобразование ReLU, которое не показано на рис. 12, является функцией активации, применяемой после каждого сверточного слоя и полносвязного слоя.

Основной задачей для предложенных моделей является оценка живого веса коровы с использованием входного RGB изображения или/и входной карты глубины. Регрессия живого веса входного изображения вычисляется с помощью функций потерь следующим образом

$$P_Y(\Theta) = \frac{1}{n} \sum_{i=1}^{n} ||F_y(X_i, \Theta) - Y_i||^2, \tag{2}$$

где Θ — набор параметров модели, X_i — входное RGB изображение или/и входная карта глубины, а n — номер обучающего набора данных. P_Y — это потеря между оцененным весом $F_y(X_i; \Theta)$ (выход полностью подключенного слоя OUT) и истинным весом Y_i. Потери минимизируются с помощью мини-пакетного градиентного спуска и алгоритма обратного распространения ошибок.

Изначально набор данных был разделен случайным образом на две части — обучающий (70%) и тестовый (30%) набор данных. Кроме того, 20% обучающего набора данных используется для валидации. Для глубокого обучения оптимизация гиперпараметров важ-

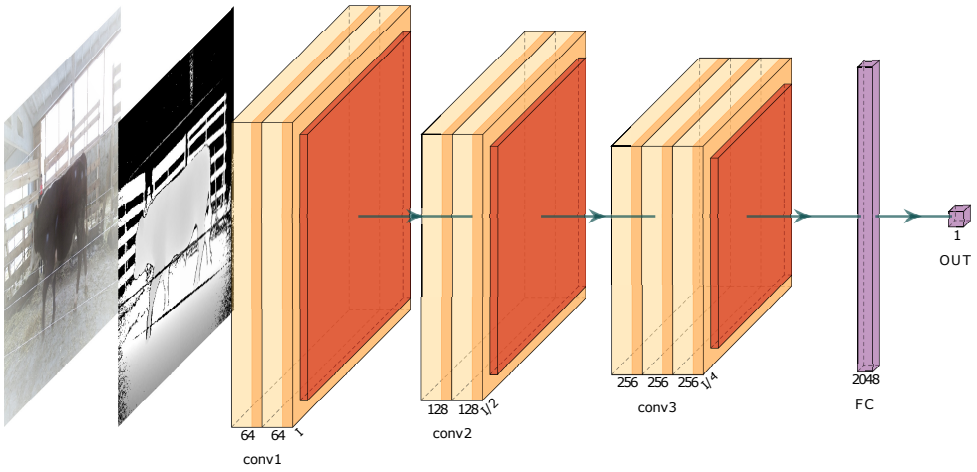


Рис. 10. Структура сверточной нейронной сети MRGBDM для прогнозирования живого веса коровы с использованием входного RGB изображения и входной карты глубины

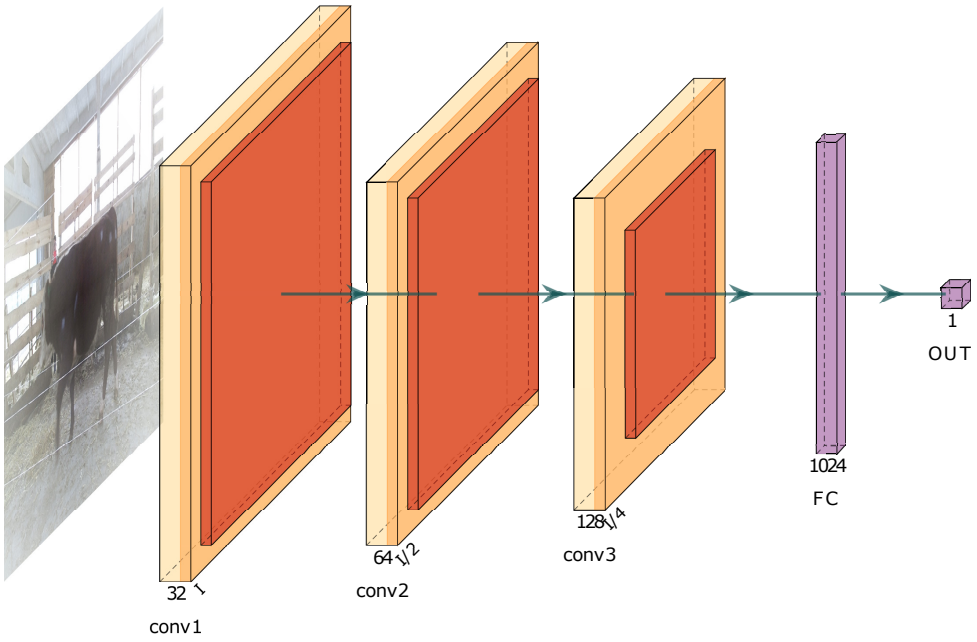


Рис. 11. Структура сверточной нейронной сети MRGB для прогнозирования живого веса коровы с использованием входного RGB изображения

на для решения проблемы выбора набора оптимальных гиперпараметров. Был использован традиционный способ оптимизации гиперпараметров с помощью поиска с кросс-валидацией (GridSearchCV [32]), который представляет собой просто исчерпывающий поиск по заданному вручную подмножеству гиперпараметрического пространства глубокой сети. С помощью GridSearchCV были найдены оптимальные гиперпараметры для всех использованных моделей.

2.1. Дополнение данных (Data augmentation)

При ограниченном количестве доступных изображений у нас мало вариативности в данных, что может привести к переобучению. Чтобы решить эту проблему, необходимо

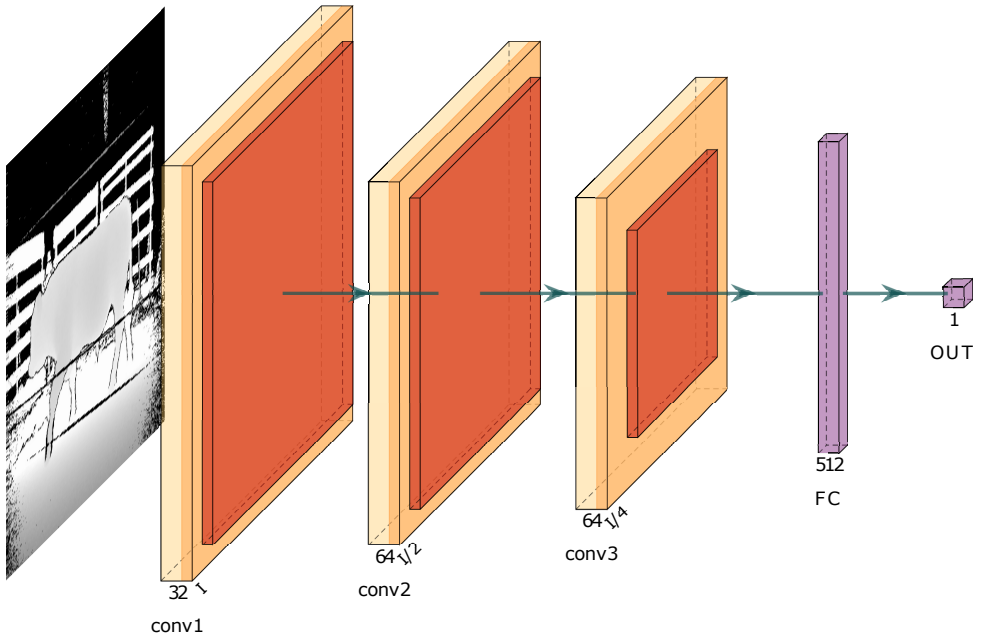


Рис. 12. Структура сверточной нейронной сети MDM для прогнозирования веса крупного рогатого скота с использованием входной карты глубины

дополнить обучающие данные синтезированными и модифицированными изображениями. Для дополнения данных использовалась комбинация преобразований поворота по трем осям X, Y, Z на ±5 градуса, сдвиг по высоте и ширине на ±50 см, что отвечает за смещение по осям X, Y, а также случайное масштабирование на ±0.2, что отвечает за смещение по оси Z. При использовании дополнения данных общий размер данных увеличился в 10 раз. В целом, первый полный набор данных состоит из 31 070 RGB изображений и 31 070 карт глубины для 154 герефордских коров. Второй полный набор данных состоит из 28 630 RGB изображений и 28 630 карт глубины для 121 абердин-ангусских коров.

2.2. Предобученные модели (Transfer Learning mode)

Модели требуют больших объемов данных для правильного и точного обучения. В области сельского хозяйства обычно трудно получить такие большие наборы данных, что связано не только с ограниченным количеством исследований, проводимых на одном предприятии, но и с объемом работы, необходимой для ручной маркировки животных. Поэтому данная работа будет исследовать подход трансферного обучения. Для этого необходимо повторно использовать предобученные модели CNN, которые ранее были обучены для других задач, и дообучить их под нашу текущую проблему. Предварительно обученная модель EfficientNet, используемая в данной работе, ранее применялась для классификации изображений [33]. В качестве экстрактора признаков используется трансфертное обучение, т.е. все слои замораживаются, и только верхний слой исходного классификатора переобучается для новых целевых классов. Кроме того, необходимо настроить нейронную сеть для достижения прогнозирования живого веса корова по RGB изображениям и картам глубины, или цветным проекциям и 2.5D проекциям. Мы переносим все веса из EfficientNet, но заменяем последний полносвязанный слой (FC8) новым последним полностью связанным слоем и слоем softmax. Новый последний слой имеет размер, равный 1, а веса инициализируют-

ся случайным образом из гауссовского распределения с нулевым средним и стандартным отклонением 0.01. Мы используем SGD с использованием мини-батчей из 32 образцов и устанавливаем скорость обучения 0.001 для предварительно обученных слоев, и скорость обучения 0.01 для последнего выходного слоя.

3. Результаты экспериментов

Для оценки эффективности моделей, используемых в данном исследовании для прогнозирования живого веса крупного рогатого скота, применялись различные критерии. Рассмотрим различные общепринятые метрики. Были использованы средняя абсолютная ошибка (MAE) и средняя абсолютная процентная ошибка (MAPE) в качестве показателей эффективности, которые определяются как

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - f_i|, \tag{3}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - f_i}{y_i}\right|, \tag{4}$$

где n — число образцов набора данных, $\bar{y}$ — среднее значение всех известных значений живого веса, y_i, $i = 1, \ldots, n$ — известное значение живого веса, и f_i, $i = 1, \ldots, n$ — предсказанное значение живого веса.

Модели обучаются на графическом процессоре Tesla V100 в течение 27 эпох. Оптимизатор — Adam, скорость обучения — 1e-5, затухание веса — 1e-5. Размер партии задается максимально возможным в пределах допустимого диапазона видеопамяти 16 Гб. Модель каждой эпохи будет тестироваться на тестовом наборе данных, и в случае, если модель имеет меньший MAE, чем на обучающей выборке, то она будет сохранена. MAE — это средняя абсолютная ошибка между предсказанным весом и истинным значением. В обучающем наборе данных 48 000 изображений, в тестовом наборе данных 11 700 изображений с крупным рогатым скотом 123 герефордовских коров и 96 абердин-ангусс коров. Тестовый набор данных состоит из 31 герефордовских коров и 25 абердин-ангусс коров, которые полностью не связаны с тренировочным набором данных.

В работе [17] были получены результаты части экспериментов, в которых на вход предложенным моделям с глубоким обучением подавались RGB изображения и карты глубины. Результаты показывают, что модели становятся лучше, когда используется дополнение данных и тонкая настройка. В данной работе мы использовали кроме RGB изображения и карты глубины еще и цветные проекции, и 2.5D проекции.

Результаты экспериментов приведены в табл. 1. Мы обучаем оригинальные сети MRGBDM, MRGB, MDM и предобученную EfficientNet (ENET), в качестве входа используем разные комбинации RGB изображений и карты глубины, а также RGB проекции и проекции карты глубины. В табл. 1 приведены результаты различных метрик, используемых для оценки эффективности модели на обучающей и тестовых наборах данных. Таблица 1 содержит итоговую точность прогнозирования живого веса крупного рогатого скота для каждой протестированной модели. Предложенная модель MRGBDM основана на применении RGB проекции и проекции карты глубины и является лучшей по точности прогноза с показателем 8.4 по метрике MAPE. Из таблицы видно, что использование RGB проекции и проекции карты может значительно уменьшить ошибки MAE и MAPE. Можно сделать

вывод, что карта глубины содержит много ценных характеристик для регрессии изображения, в отличие от RGB изображения. Предобученная ENET дает хуже результаты, MAPE меньше на 1.5, чем у лучшей модели MRGBDM с MAPE 8.4.

Таблица 1. Результаты прогнозирования живого веса MAE и MAPE крупного рогатого скота с помощью предложенных моделей MRGBDM, MRGB, MDM и предварительно обученной моделью EfficientNet (ENET) на обучающем и тестовом наборе данных

Вход	Модель	Обучающая		Тестовая	
		MAE	MAPE	MAE	MAPE
Сырые RGB изображения и карты глубины	MRGBDM	37.9	9.1	40.1	9.6
	MRGB	46.9	11.1	50.3	11.9
	MDM	40.5	9.5	43.5	10.2
	ENET	41.1	9.8	43.6	10.4
RGB проекция и проекции карты глубины	MRGBDM	34.2	8.1	35.5	8.4
	MRGB	42.5	10.1	45.6	10.8
	MDM	37.6	8.9	39.7	9.4
	ENET	38.9	9.2	41.8	9.9

Бесконтактное измерение веса позволяет сэкономить время и избежать стресса у крупного рогатого скота. Лучшим вариантом является измерение веса по RGB изображениям и карте глубины с пространственной информацией. Были использованы для экспериментов две базы с крупным рогатым скотом 154 герефордовских коров и 121 абердин-ангусс коров. Также был использован алгоритм выбора кадров, на которых присутствует целое животное в правильной осанке, из общей последовательности изображений с помощью метода детектирования области головы, бедра и тела животного на двухмерном RGB изображении. Однако, существует ряд проблем: вариативность окружающей среды, шумы и отсутствие части данных, положение в пространстве животного, общий масштаб, небольшая выборка обучающих данных, что необходимо учитывать особенно для обучения глубоких сетей. Поэтому мы разработали ряд алгоритмов предварительной обработки RGB изображений и карты глубины, включая, подавление шума на RGB изображениях и карт глубины, восстановление облака точек, а также удаление фона из облака точек и нормализация позы животного, что позволило сохранить информацию о теле крупного рогатого скота и исключить влияние окружающей среды. После обработки изображения поступают в модель прогнозирования веса. В данной статье было предложено использовать RGB проекцию и проекцию карты глубины в качестве входа в глубокую нейронную сеть вместо сырых RGB изображений и карт глубины, что позволило повысить надежность прогнозирования живого веса крупного рогатого скота. Кроме того, с помощью проекций удалось получить трехмерное дополнение данных, что позволило расширить размер выборки для обучения глубоких сетей.

В будущем для повышения точности регрессии изображений можно использовать предварительно обработанное облако точек в качестве входа в глубокую нейронную сеть.

Бесконтактное измерение живого веса крупного рогатого скота может быть использовано в сельском хозяйстве: для объективной оценки племенных животных в ходе бонитировки; для оценки коммерческой стоимости скота при работе аукционов скота разных стран; для обоснования дальнейшего использования молодняка, в том числе для откорма с

перспективой исключения необходимости выполнения генетической экспертизы животных; для разработки аналоговой технологии оценки состояния здоровья и продуктивности животных на промышленных птицеводческих и свиноводческих комплексах. Потребителями созданного интеллектуального продукта могут стать: 1) ассоциации по породам скота и союзы, занимающиеся разведением чистопородных животных; 2) аукционы и рынки живого скота; 3) рестораторы и магазины, приобретающие животноводческую продукцию; 4) откормочные площадки и другие организации, проводящие экспертизу скота.

Заключение

В данной работе ыло изучено применение глубоких моделей к задаче регрессии изображений для прогнозирования живого веса крупного рогатого скота. Были предложены методы для предобработки RGB изображений и карты глубины и создании цветной RGB проекции и 2.5D карты глубины для прогнозирования живого веса на основе регрессии изображений с помощью методов глубокого обучения. Кроме того, был использован метод трехмерной аугментации цветной проекции и 2.5D карты глубины с помощью жестких преобразований в виде трехмерных вращений и перемещений, что позволило увеличить ограниченный набор данных и повысить эффективность прогнозирования живого веса при наличии вариаций позы, положения и масштаба животного. Была получена оценка эффективности предложенных моделей MRGBDM, MRGB, MDM и предварительно обученную модель ENET с помощью методов тонкой настройки и дополнения данных. Результаты показывают, что модели становятся эффективней, когда используется дополнение данных и тонкая настройка. Лучшей моделью является предложенная модель MRGBDM с MAPE 8.4%, использующая цветную проекцию и 2.5D карты глубины. Можно сделать вывод, что карта глубины содержит много ценных характеристик для регрессии изображений, в отличие от RGB изображения. Были показаны результаты на реальных наборах данных, которые демонстрируют, что предложенная модель MRGBDM может достичь уровня точности измерения веса, сравнимого с тем, который достигается традиционным взвешиванием. В будущем предполагается предобработать карту глубины, чтобы выбрать только область с животным. Также в будущем для повышения точности регрессии изображений можно использовать облако точек в качестве входа в глубокую нейронную сеть.

Литература

1. Wang Z., Shadpour S., Chan E., *et al.* ASAS-NANP SYMPOSIUM: Applications of machine learning for livestock body weight prediction from digital images // Journal of Animal Science. 2021. Vol. 99, no. 2. DOI: 10.1093/jas/skab022.
2. Ruchay A., Kober V., Dorofeev K., *et al.* Accurate body measurement of live cattle using three depth cameras and non-rigid 3-D shape recovery // Computers and Electronics in Agriculture. 2020. Vol. 179. P. 105821. DOI: 10.1016/j.compag.2020.105821.
3. Kuzuhara Y., Kawamura K., Yoshitoshi R., *et al.* A preliminarily study for predicting body weight and milk properties in lactating Holstein cows using a three-dimensional camera system // Computers and Electronics in Agriculture. 2015. Vol. 111. P. 186–193. DOI: 10.1016/j.compag.2014.12.020.

4. Sawanon S., Boonsaen P., Innurak P. Body Measurements of Male Kamphaeng Saen Beef Cattle as Parameters for Estimation of Live Weight // Kasetsart Journal - Natural Science. 2011. Vol. 45, no. 3. P. 428–434.

5. Wangchuk K., Wangdi J., Mindu M. Comparison and reliability of techniques to estimate live cattle body weight // Journal of Applied Animal Research. 2017. Vol. 46. P. 4. DOI: 10.1080/09712119.2017.1302876.

6. Vanvanhossou F., Diogo R., Dossa L. Estimation of live bodyweight from linear body measurements and body condition score in the West African Savannah Shorthorn Cattle in North-West Benin // Cogent Food And Agriculture. 2018. Vol. 4, no. 1. P. 1549767. DOI: 10.1080/23311932.2018.1549767.

7. Huma Z., Iqbal F. Predicting the body weight of Balochi sheep using a machine learning approach // Turkish journal of veterinary and animal sciences. 2019. Vol. 43, no. 4. P. 500–506. DOI: 10.3906/vet-1812-23.

8. Hempstalk K., Mcparland S., Berry D. Machine learning algorithms for the prediction of conception success to a given insemination in lactating dairy cows // Journal of dairy science. 2015. Vol. 98, no. 8. P. 5262–5273. DOI: 10.3168/jds.2014-8984.

9. Miller G.A., Hyslop J.J., Barclay D., *et al.* Using 3D Imaging and Machine Learning to Predict Liveweight and Carcass Characteristics of Live Finishing Beef Cattle // Frontiers in Sustainable Food Systems. 2019. Vol. 3. P. 30. DOI: 10.3389/fsufs.2019.00030.

10. Milosevic B., Ciric S., Lalic N., *et al.* Machine learning application in growth and health prediction of broiler chickens // World's Poultry Science Journal. 2019. Vol. 75. P. 401–410. DOI: 10.1017/S0043933919000254.

11. Weber V., Weber F., Gomes R., *et al.* Prediction of Girolando cattle weight by means of body measurements extracted from images // Revista Brasileira de Zootecnia. 2020. Mar. Vol. 49. DOI: 10.37496/rbz4920190110.

12. Tasdemir S., Urkmez A., Inal S. Determination of body measurements on the Holstein cows using digital image analysis and estimation of live weight with regression analysis // Computers and Electronics in Agriculture. 2011. Vol. 76, no. 2. P. 189–197. DOI: 10.1016/j.compag.2011.02.001.

13. Pezzuolo A., Milani V., Zhu D., *et al.* On-Barn Pig Weight Estimation Based on Body Measurements by Structure-from-Motion (SfM) // Sensors. 2018. Vol. 18, no. 11. Article 3603. DOI: 10.3390/s18113603.

14. Song X., Bokkers E., Tol P. van der, *et al.* Automated body weight prediction of dairy cows using 3-dimensional vision // Journal of Dairy Science. 2018. Vol. 101, no. 5. P. 4448–4459. DOI: 10.3168/jds.2017-13094.

15. Ranganathan H., Venkateswara H., Chakraborty S., Panchanathan S. Deep Active Learning for Image Regression // Deep Learning Applications. Singapore: Springer Singapore, 2020. P. 113–135. DOI: 10.1007/978-981-15-1816-4_7.

16. Bezsonov O., Lebediev O., Lebediev V., *et al.* Breed Recognition and Estimation of Live Weight of Cattle Based on Methods of Machine Learning and Computer Vision // Eastern-European Journal of Enterprise Technologies. 2021. Vol. 6/9, no. 114. P. 64–74. DOI: 10.15587/1729-4061.2021.247648.

17. Ruchay A., Dorofeev K., Kalschikov V., *et al.* Live weight prediction of cattle using deep image regression // 2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor). 2021. P. 32–36. DOI: 10.1109/MetroAgriFor52389.2021.9628547.

18. Ruchay A., Dorofeev K., Kober A., *et al.* Accuracy analysis of 3D object shape recovery using depth filtering algorithms // Applications of Digital Image Processing XLI. Vol. 10752. SPIE, 2018. P. 1075221–10. DOI: 10.1117/12.2319907.

19. Ruchay A., Kolpakov V., Kosyan D., *et al.* Genome-Wide Associative Study of Phenotypic Parameters of the 3D Body Model of Aberdeen Angus Cattle with Multiple Depth Cameras // Animals. 2022. Vol. 12, no. 16. Article 2128. DOI: 10.3390/ani12162128.

20. Lu J., Guo H., Du A., *et al.* 2-D/3-D fusion-based robust pose normalisation of 3-D livestock from multiple RGB-D cameras // Biosystems Engineering. 2021. Vol. 223. P. 129–141. DOI: 10.1016/j.biosystemseng.2021.12.013.

21. Bochkovskiy A., Wang C.-Y., Liao H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection // CoRR. 2020. Vol. abs/2004.10934. arXiv: 2004.10934. URL: https://arxiv.org/abs/2004.10934.

22. Hu Y., Luo X., Gao Z., *et al.* Curve Skeleton Extraction from Incomplete Point Clouds of Livestock and Its Application in Posture Evaluation // Agriculture. 2022. Vol. 12, no. 7. Article 998. DOI: 10.3390/agriculture12070998.

23. Ruchay A., Kober V. Clustered impulse noise removal from color images with spatially connected rank filtering // Applications of Digital Image Processing XXXIX. Vol. 9971. SPIE, 2016. 99712Y–10. DOI: 10.1117/12.2236785.

24. Ruchay A., Kober V. Removal of impulse noise clusters from color images with local order statistics // Applications of Digital Image Processing XL. Vol. 10396. SPIE, 2017. P. 1039626–10. DOI: 10.1117/12.2272718.

25. Ruchay A., Kober V. Impulsive noise removal from color video with morphological filtering // Applications of Digital Image Processing XL. Vol. 10396. SPIE, 2017. P. 1039627–9. DOI: 10.1117/12.2272719.

26. Ruchay A., Kober V. Impulsive Noise Removal from Color Images with Morphological Filtering // Analysis of Images, Social Networks and Texts. Vol. 10716. Cham: Springer International Publishing, 2018. P. 280–291. Lecture Notes in Computer Science. DOI: 10.1007/978-3-319-73013-4_26.

27. Ruchay A., Dorofeev K., Kalschikov V. A novel switching bilateral filtering algorithm for depth map // Computer Optics. 2019. Vol. 43, no. 6. P. 1001–1007. DOI: 10.18287/2412-6179-2019-43-6-1001-1007.

28. Ruchay A.N., Dorofeev K.A., Kalschikov V.. Accuracy analysis of 3D object reconstruction using point cloud filtering algorithms // CEUR Workshop Proceedings. 2019. Vol. 2391. P. 169–174. DOI: 10.18287/1613-0073-2019-2391-169-174.

29. Rusu R.B., Cousins S. 3D is here: Point Cloud Library (PCL) // 2011 IEEE International Conference on Robotics and Automation. 2011. P. 1–4.

30. Ruchay A., Gladkov A., Chelabiev R. Fast 3D object pose normalization for point cloud // Applications of Digital Image Processing XLIV. Vol. 11842. SPIE, 2021. DOI: 10.1117/12.2593893.

31. Ruchay A., Kalschikov V., Gridnev A., Guo H. Fast 3D object symmetry detection for point cloud // Applications of Digital Image Processing XLIV. Vol. 11842. SPIE, 2021. DOI: 10.1117/12.2593895.

32. Chollet F. *et al.* Keras. 2015. URL: https://github.com/fchollet/keras.

33. Tan M., Le Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks // Proceedings of the 36th International Conference on Machine Learning. Vol. 97 / ed. by K. Chaudhuri, R. Salakhutdinov. PMLR, 2019. P. 6105–6114. Proceedings of Machine Learning Research.

Ручай Алексей Николаевич, к.ф.-м.н., доцент, заведующий кафедрой компьютерной безопасности и прикладной алгебры; Челябинский государственный университет (Челябинск, Российская Федерация); доцент кафедры защиты информации, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация); старший научный сотрудник, Федеральный научный центр биологических систем и агротехнологий РАН (Оренбург, Российская Федерация).

DOI: 10.14529/cmse230101

PREDICTION MODEL OF LIVE WEIGHT USING DEEP REGRESSION RGB-D IMAGES

© 2023 A.N. Ruchay[1,2,3]

[1]*South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia),*
[2]*Chelyabinsk State University (st. Br. Kashirinyh 129, Chelyabinsk, 454001 Russia),*
[3]*Federal Research Centre of Biological Systems and Agrotechnologies of RAS (st. 9 Yanvarya 29, Orenburg, 460000 Russia)*
E-mail: ran@csu.ru
Received: 13.02.2023

Predicting live weight helps to monitor animal health, effectively conduct genetic selection and determine optimal time of slaughter. On large farms, accurate and expensive industrial scales are used to measure live weight. However, a promising alternative to that is estimation of live weight by using morphometric measurements of an animal and then applying regression equations linking such measurements to live weight. Manual measurements of animals using a tape measure are time-consuming and stressful for animals. Therefore, computer vision technology is now increasingly being used for non-contact morphometric measurements. This article proposes a new model for predicting live weight based on image regression using deep learning techniques. It is shown that, on real datasets the proposed model achieves weight measurement accuracy with a MAE of 35.5 and MAPE of 8.4 on the test dataset.

Keywords: image regression, live body weight prediction, cattle, deep learning.

FOR CITATION

Ruchay A.N. Prediction Model of Live Weight Using Deep Regression RGB-D Images. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2023. Vol. 12, no. 1. P. 5–27. (in Russian) DOI: 10.14529/cmse230101.

References

1. Wang Z., Shadpour S., Chan E., *et al.* ASAS-NANP SYMPOSIUM: Applications of machine learning for livestock body weight prediction from digital images. Journal of Animal Science. 2021. Vol. 99, no. 2. DOI: 10.1093/jas/skab022.
2. Ruchay A., Kober V., Dorofeev K., *et al.* Accurate body measurement of live cattle using three depth cameras and non-rigid 3-D shape recovery. Computers and Electronics in Agriculture. 2020. Vol. 179. P. 105821. DOI: 10.1016/j.compag.2020.105821.
3. Kuzuhara Y., Kawamura K., Yoshitoshi R., *et al.* A preliminarily study for predicting body weight and milk properties in lactating Holstein cows using a three-dimensional camera system. Computers and Electronics in Agriculture. 2015. Vol. 111. P. 186–193. DOI: 10.1016/j.compag.2014.12.020.
4. Sawanon S., Boonsaen P., Innurak P. Body Measurements of Male Kamphaeng Saen Beef Cattle as Parameters for Estimation of Live Weight. Kasetsart Journal - Natural Science. 2011. Vol. 45, no. 3. P. 428–434.
5. Wangchuk K., Wangdi J., Mindu M. Comparison and reliability of techniques to estimate live cattle body weight. Journal of Applied Animal Research. 2017. Vol. 46. P. 4. DOI: 10.1080/09712119.2017.1302876.
6. Vanvanhossou F., Diogo R., Dossa L. Estimation of live bodyweight from linear body measurements and body condition score in the West African Savannah Shorthorn Cattle in North-West Benin. Cogent Food And Agriculture. 2018. Vol. 4, no. 1. P. 1549767. DOI: 10.1080/23311932.2018.1549767.
7. Huma Z., Iqbal F. Predicting the body weight of Balochi sheep using a machine learning approach. Turkish journal of veterinary and animal sciences. 2019. Vol. 43, no. 4. P. 500–506. DOI: 10.3906/vet-1812-23.
8. Hempstalk K., Mcparland S., Berry D. Machine learning algorithms for the prediction of conception success to a given insemination in lactating dairy cows. Journal of dairy science. 2015. Vol. 98, no. 8. P. 5262–5273. DOI: 10.3168/jds.2014-8984.
9. Miller G.A., Hyslop J.J., Barclay D., *et al.* Using 3D Imaging and Machine Learning to Predict Liveweight and Carcass Characteristics of Live Finishing Beef Cattle. Frontiers in Sustainable Food Systems. 2019. Vol. 3. P. 30. DOI: 10.3389/fsufs.2019.00030.
10. Milosevic B., Ciric S., Lalic N., *et al.* Machine learning application in growth and health prediction of broiler chickens. World's Poultry Science Journal. 2019. Vol. 75. P. 401–410. DOI: 10.1017/S0043933919000254.
11. Weber V., Weber F., Gomes R., *et al.* Prediction of Girolando cattle weight by means of body measurements extracted from images. Revista Brasileira de Zootecnia. 2020. Mar. Vol. 49. DOI: 10.37496/rbz4920190110.

12. Tasdemir S., Urkmez A., Inal S. Determination of body measurements on the Holstein cows using digital image analysis and estimation of live weight with regression analysis. Computers and Electronics in Agriculture. 2011. Vol. 76, no. 2. P. 189–197. DOI: 10.1016/j.compag.2011.02.001.

13. Pezzuolo A., Milani V., Zhu D., *et al.* On-Barn Pig Weight Estimation Based on Body Measurements by Structure-from-Motion (SfM). Sensors. 2018. Vol. 18, no. 11. Article 3603. DOI: 10.3390/s18113603.

14. Song X., Bokkers E., Tol P. van der, *et al.* Automated body weight prediction of dairy cows using 3-dimensional vision. Journal of Dairy Science. 2018. Vol. 101, no. 5. P. 4448–4459. DOI: 10.3168/jds.2017-13094.

15. Ranganathan H., Venkateswara H., Chakraborty S., Panchanathan S. Deep Active Learning for Image Regression. Deep Learning Applications. Singapore: Springer Singapore, 2020. P. 113–135. DOI: 10.1007/978-981-15-1816-4_7.

16. Bezsonov O., Lebediev O., Lebediev V., *et al.* Breed Recognition and Estimation of Live Weight of Cattle Based on Methods of Machine Learning and Computer Vision. Eastern-European Journal of Enterprise Technologies. 2021. Vol. 6/9, no. 114. P. 64–74. DOI: 10.15587/1729-4061.2021.247648.

17. Ruchay A., Dorofeev K., Kalschikov V., *et al.* Live weight prediction of cattle using deep image regression. 2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor). 2021. P. 32–36. DOI: 10.1109/MetroAgriFor52389.2021.9628547.

18. Ruchay A., Dorofeev K., Kober A., *et al.* Accuracy analysis of 3D object shape recovery using depth filtering algorithms. Applications of Digital Image Processing XLI. Vol. 10752. SPIE, 2018. P. 1075221–10. DOI: 10.1117/12.2319907.

19. Ruchay A., Kolpakov V., Kosyan D., *et al.* Genome-Wide Associative Study of Phenotypic Parameters of the 3D Body Model of Aberdeen Angus Cattle with Multiple Depth Cameras. Animals. 2022. Vol. 12, no. 16. Article 2128. DOI: 10.3390/ani12162128.

20. Lu J., Guo H., Du A., *et al.* 2-D/3-D fusion-based robust pose normalisation of 3-D livestock from multiple RGB-D cameras. Biosystems Engineering. 2021. Vol. 223. P. 129–141. DOI: 10.1016/j.biosystemseng.2021.12.013.

21. Bochkovskiy A., Wang C.-Y., Liao H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. CoRR. 2020. Vol. abs/2004.10934. arXiv: 2004.10934. URL: https://arxiv.org/abs/2004.10934.

22. Hu Y., Luo X., Gao Z., *et al.* Curve Skeleton Extraction from Incomplete Point Clouds of Livestock and Its Application in Posture Evaluation. Agriculture. 2022. Vol. 12, no. 7. Article 998. DOI: 10.3390/agriculture12070998.

23. Ruchay A., Kober V. Clustered impulse noise removal from color images with spatially connected rank filtering. Applications of Digital Image Processing XXXIX. Vol. 9971. SPIE, 2016. 99712Y–10. DOI: 10.1117/12.2236785.

24. Ruchay A., Kober V. Removal of impulse noise clusters from color images with local order statistics. Applications of Digital Image Processing XL. Vol. 10396. SPIE, 2017. P. 1039626–10. DOI: 10.1117/12.2272718.

25. Ruchay A., Kober V. Impulsive noise removal from color video with morphological filtering. Applications of Digital Image Processing XL. Vol. 10396. SPIE, 2017. P. 1039627–9. DOI: 10.1117/12.2272719.

26. Ruchay A., Kober V. Impulsive Noise Removal from Color Images with Morphological Filtering. Analysis of Images, Social Networks and Texts. Vol. 10716. Cham: Springer International Publishing, 2018. P. 280–291. Lecture Notes in Computer Science. DOI: 10.1007/978-3-319-73013-4_26.

27. Ruchay A., Dorofeev K., Kalschikov V. A novel switching bilateral filtering algorithm for depth map. Computer Optics. 2019. Vol. 43, no. 6. P. 1001–1007. DOI: 10.18287/2412-6179-2019-43-6-1001-1007.

28. Ruchay A.N., Dorofeev K.A., Kalschikov V.. Accuracy analysis of 3D object reconstruction using point cloud filtering algorithms. CEUR Workshop Proceedings. 2019. Vol. 2391. P. 169–174. DOI: 10.18287/1613-0073-2019-2391-169-174.

29. Rusu R.B., Cousins S. 3D is here: Point Cloud Library (PCL). 2011 IEEE International Conference on Robotics and Automation. 2011. P. 1–4.

30. Ruchay A., Gladkov A., Chelabiev R. Fast 3D object pose normalization for point cloud. Applications of Digital Image Processing XLIV. Vol. 11842. SPIE, 2021. DOI: 10.1117/12.2593893.

31. Ruchay A., Kalschikov V., Gridnev A., Guo H. Fast 3D object symmetry detection for point cloud. Applications of Digital Image Processing XLIV. Vol. 11842. SPIE, 2021. DOI: 10.1117/12.2593895.

32. Chollet F. *et al.* Keras. 2015. URL: https://github.com/fchollet/keras.

33. Tan M., Le Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning. Vol. 97 / ed. by K. Chaudhuri, R. Salakhutdinov. PMLR, 2019. P. 6105–6114. Proceedings of Machine Learning Research.

DOI: 10.14529/cmse230102

A METHOD FOR CREATING STRUCTURAL MODELS OF TEXT DOCUMENTS USING NEURAL NETWORKS

© 2023 D.V. Berezkin, I.A. Kozlov, P.A. Martynyuk, A.M. Panfilkin

Bauman Moscow State Technical University
(st. 2nd Baumanskaya 5/1, Moscow, 105005 Russian Federation)
E-mail: berezkind@bmstu.ru, kozlovilya89@gmail.com,
martapauline@yandex.ru, panfilkinam@student.bmstu.ru
Received: 03.11.2022

The article describes modern neural network BERT-based models and considers their application for Natural Language Processing tasks such as question answering and named entity recognition. The article presents a method for solving the problem of automatically creating structural models of text documents. The proposed method is hybrid and is based on jointly utilizing several NLP models. The method builds a structural model of a document by extracting sentences that correspond to various aspects of the document. Information extraction is performed by using the BERT Question Answering model with questions that are prepared separately for each aspect. The answers are filtered via the BERT Named Entity Recognition model and used to generate the contents of each field of the structural model. The article proposes two algorithms for field content generation: Exclusive answer choosing algorithm and Generalizing answer forming algorithm, that are used for short and voluminous fields respectively. The article also describes the software implementation of the proposed method and discusses the results of experiments conducted to evaluate the quality of the method.

Keywords: information extraction, neural network, named entity recognition, question-answering system.

FOR CITATION

Berezkin D.V., Kozlov I.A., Martynyuk P.A., Panfilkin A.M. A Method for Creating Structural Models of Text Documents Using Neural Networks. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2023. Vol. 12, no. 1. P. 28–45. DOI: 10.14529/cmse230102.

Introduction

Modern information systems accumulate and process huge volumes of heterogeneous data, a significant proportion of which are text documents. Such documents are used as an input for many Natural Language Processing (NLP) tasks that have seen a significant progress in recent years, mostly due to the development of deep learning technologies.

Many NLP tasks require comparing two text documents. Such tasks include text clustering (which requires computing the similarity between two documents in order to determine if they can be put in the same cluster), information retrieval (which involves determining how close a document is to a user's query), plagiarism detection and more. In most cases, comparison of the documents takes into account whole texts of both documents. A typical implementation of such a comparison involves representing the entire document using a vector model (such as Bag of Words, TF-IDF, Word2Vec or other embedding models) and comparing vector models of two documents via various similarity measures such as Cosine Similarity or Word Mover's Distance.

However, in some specific tasks, when comparing documents, only fragments of their texts should be taken into account. Here are some of the possible scenarios in which text documents comparison needs to be performed this way:

1. Comparison of several scientific articles on the same problem in order to determine the most efficient solution. In this case, the articles should be compared in terms of the efficiency of the methods presented.
2. Comparison of several consecutive versions of an official document regulating a certain area (for instance, a national strategy of AI development) in order to track the development of technologies that are used to achieve the goals set in the document.

The task of partial comparison can be easily performed on structured documents when each document is represented by a frame where the fragment of interest is located in a separate field. However, in case of unstructured documents this task is challenging: the respective fragments may be located in different parts of documents and may be worded using different terms. A possible solution to this problem consists of building a structural model of every document and then comparing the models instead of documents themselves. A structural model of a text document is a frame, each element of which corresponds to a certain aspect of the document and each value of an element is a fragment of the text in the document. By "aspect" of the document we mean a semantic component of the text in the document, corresponding to a certain query. For instance, in the case of choosing an article that provides the most efficient solution to a certain problem, the aspect of interest is "Efficiency of the solution" and it can be described by a query "How good is the quality of the proposed method?"

When analyzing scientific articles, possible aspects to extract are the title and authors of the article, the goal and relevance of the research, existing and proposed models and methods. When analyzing national strategies in technical areas, we are interested in other aspects such as the expected time of the strategy's implementation and technologies that are used to achieve the strategy's goals.

A structural model of a text document is created by detecting fragments of the text in the document that corresponds to the aspects of interest. Creation of a structural model can be considered a special case of a general Information Extraction (IE) task which consists of extracting structured information from unstructured natural language text documents. The article considers the solution of this problem using modern neural network technologies.

The article is organized as follows. Section 1 presents the formulation of the problem of creating a structural model of a document as a special case of a general Information Extraction task. Section 2 describes existing approaches to Information Extraction task. Section 3 is devoted to the proposed method and describes its steps and technologies used in each of the steps. In Section 4 we present a software implementation of the proposed method. Section 5 describes experiments conducted to evaluate the quality of the proposed method. Conclusion contains a brief summary of the results obtained in the work and directions for further research.

1. Formulation of the Problem

The general Information Extraction problem can be formulated as determining a set of frame instances $F = \{f_i\}$ and a relation $R_F \subseteq (F \times S \times V)$ based on a set of text documents $D = \{d_i\}$. Each frame instance f_i represents a certain entity extracted from text documents. The relation R_F determines values V of slots S of frame instances F. The slot values V are the fragments extracted from the text documents D.

Various specific Information Extraction tasks fit into this formulation and differ from each other in terms of what objects are represented with frames. Examples of such tasks are the extraction of named entities (names, organizations and geographical locations) [1], addresses [2]

and events [3]. In each of these cases, it is assumed that the analyzed documents D and frame instances F are related in a one-to-many way as multiple entities may be extracted from each document. However, the task that is considered in this article requires that for each document one and only one frame instance is formed — a frame instance that describes the document in a structured way. The slots of this frame instance should correspond to aspects of the document that are of interest to a specific task that is being solved. In this regard, we present a more specific formulation of the problem, taking into account the described requirements.

The task is to determine a set of structural models $M = \{m_i\}$, $i = \overline{1..N_d}$, based on a set of text documents $D = \{d_i\}$, where m_i is a structural model of the document d_i, N_d is the number of documents (as well as the number of structural models). Each structural model m_i is a tuple $m_i = (m_i^j)$, $j = \overline{1..N_a}$, where an element m_i^j is a text string that describes the j-th aspect of the document d_i, and N_a is the number of aspects of interest. Each text string m_i^j is a fragment extracted from the text of the document d_i. Further, we will also use terms “card” and “fields” to denote the structural model and its elements respectively.

2. Related Work

Traditionally, Information Extraction tasks are solved mainly using two approaches: rule-based and probabilistic. However, due to the rapid development of neural networks in recent years, they have found application in solving various problems of text analysis, including the problem of Information Extraction. In this section, we will consider both traditional approaches and the neural network approach.

2.1. Rule-based Approach

Rule-based Information Extraction methods use extraction rules written in a formal language. An extraction rule imposes a set of restrictions on the analyzed text fragment. These restrictions may apply to orthographic, morphological, syntactic and semantic features of separate words, as well as to relations between them. If the text fragment meets the rule's restrictions, it is concluded that this fragment contains the sought-for entity. In this case, a new frame instance is created that represents the extracted entity. Its slots are initialized with some elements of the text fragment.

Depending on the formal language that is used for writing rules, rule-based Information Extraction methods can be divided into two categories: propositional and relational [4]. Propositional methods use rules that are written in the language of zero order (propositional) logic. Expressions in such a language can only include attributes of words and phrases. The most common attributes are morphological features of words, syntactic roles of words in a sentence, and semantic classes. Relational methods use rules that are written in the language of first order logic. In addition to attributes of words and phrases, such rules can describe relations between them. The most common types of such relations are syntactic and order relations that specify the syntactic structure of a sentence and the order of phrases within a sentence respectively.

The rules can be written by an expert or generated automatically based on a set of training examples [5]. Training examples are manually tagged by experts and then are used by the Information Extraction system to infer extraction rules by generalizing restrictions during the learning process.

A rule-based approach CLIEL proposed in [6] consists of two stages of processing: organizing the text in an accessible form and subsequent extraction of information. The basis of the

mechanism is the recognition of the document layout and the use of a set of grammatical rules to extract information from commercial law documents. An approach proposed in the paper [7] also takes into account the structure of the document and uses rules to extract information. First, the structure of the document is revealed to determine what information should be extracted from its individual parts. In order to search for the relevant part of the text to extract specific information, a compact lexical dictionary is used. Second, the text is normalized, tokenized, tagged using POS Tagger, and information is extracted using templates.

2.2. Probabilistic Approach

Probabilistic Information Extraction methods are based on the construction of probabilistic models that include observable and hidden variables. Observable variables X correspond to various features of the analyzed text fragment. Hidden variables Y match elements of the text fragment to the slots of the frame that represents the entity that is being extracted. When analyzing a certain text fragment described by features x, the values of hidden variables y are determined by maximizing the conditional probability $\mathbf{P}(Y = y|X = x)$. If the value of this probability exceeds a certain threshold, a new frame instance is created and its slots are filled with elements of the text fragment in accordance with y. The probabilistic models are trained by estimating their parameters using the maximum likelihood method.

Probabilistic Information Extraction methods can use generative or discriminative models. Generative models are based on calculating the joint probability $\mathbf{P}(Y = y, X = x)$ that is then used to determine the probability $\mathbf{P}(Y = y|X = x)$. They include, among others, Naive Bayes Classifier [8] and Hidden Markov Model [9]. Discriminative models allow to directly determine the desired conditional probability $\mathbf{P}(Y = y|X = x)$. These include Conditional Random Fields [10].

The paper [11] analyzes several probabilistic models that have proven to be particularly useful for various tasks of extracting meaning from natural language texts. Most prominent among them are Hidden Markov models (HMMs), stochastic context-free grammars (SCFG), and maximal entropy (ME).

2.3. Neural Network Approach

The use of traditional methods requires a large amount of linguistic resources: tagged corpus of texts, dictionaries, thesauri. In order to take into account all the ways in which the aspects of interest can be described in documents, it is necessary to prepare a large number of rules, which requires a lot of expert work and a high level of knowledge in the subject area. These problems can be avoided via the usage of methods based on modern neural network technologies due to the ability of neural networks to independently determine the feature space when processing the training corpus of text documents. This technology is called representation learning. It aims to automatically obtain informative representations of objects from raw data. Hence, preliminary training of modern language models does not require a detailed tagging of text elements with features (morphological, syntactic, semantic, and others). As a result of training, the neural network is able to represent every word with a vector of numbers (so-called “word embedding”), each of which is a value of some feature selected by the network. Unlike traditional vector models (such as “bag of words” and TF-IDF), embeddings reflect the semantics of words, and also allow to evaluate their contextual proximity. Deep learning models trained to form word embeddings are called pre-trained. The pre-trained models are universal and are used to solve NLP problems in various subject areas. To solve a specific NLP problem, a pre-trained model needs fine-tuning,

which requires significantly less data than pre-training. At the moment, ready-to-use datasets are available for fine-tuning and testing neural network models. Datasets for NLP tasks are usually taken from the collections of the GLUE benchmark [12].

One of the latest and most ambitious developments in the field of neural network language models is the BERT model, released in 2018 by Google [13]. Due to the architectural features of this model, BERT is able to take into account the bidirectional context of words when pre-training representations, which gives it an advantage over other neural network language models such as Word2Vec and GloVe [14]. Thus, the model creates different embeddings for homonymous words, which allows avoiding false interpretations of words in further work with text documents.

BERT model is based on the Transformer architecture and uses the Self-Attention mechanism, which makes it easy to adapt the model for solving specific NLP tasks by fine-tuning it on task-relevant input and output data [13]. Fine-tuned BERT-based neural network models have been used for solving various NLP tasks including the Information Extraction problem. For instance, the DeepPavlov open source library contains special purpose BERT models configured to solve the problem of Named Entity Recognition (NER) [15]. Also, the DeepPavlov library contains a BERT model fine-tuned for extracting relations between objects. Named Entity Recognition and relation extraction are subtasks of the Information Extraction problem that have been widely studied by researchers from the perspective of possible use of modern neural network models [16]. However, the existing methods that have been proposed in this area are highly specialized and cannot be directly applied to solve the task of building structural models of text documents.

There has been a tendency to solve various NLP problems by converting them to the Question-Answering (Q&A) task [17]. For example, the paper [18] proposes using a Q&A model for extracting named entities. Paper [19] describes multi-turn Q&A method for relation extracting using question templates. Authors use BERT model as a backbone for the Q&A framework. Paper [20] implements information extraction system based on the functioning of a Q&A model proposed by authors and named QA4IE. Since Q&A-based approach has been proven effective for solving IE tasks, we decided to use it as a basis for our method for creating structural models of text documents.

3. The Proposed Method for Creating Structural Models of Text Documents

In order to build structural models of documents we use BERT based neural network models. One of them is the Q&A model. It is fine-tuned on the SQuAD (Stanford Question Answering Dataset) dataset, which consists of questions based on Wikipedia articles, where the answer to each question is a text fragment of an article [21]. Another model that we use is the NER model fine-tuned on the OntoNotes dataset, containing marked-up data from various sources (web blogs, telephone conversations, news feeds) and supporting 18 entity categories [22]. Both Q&A and NER models are open source and ready to use: BERT Q&A is provided in the official Google Research github repository [23] and BERT NER can be obtained from the DeepPavlov open source library [24]. In addition, we use the Sentence-BERT [25, 26] model that generates sentence embeddings, in contrast to the usual BERT embeddings that are formed for tokens — words or parts of words contained in the model dictionary. All models are already pre-trained and fine-tuned, they do not require additional training. All of them are generic English language

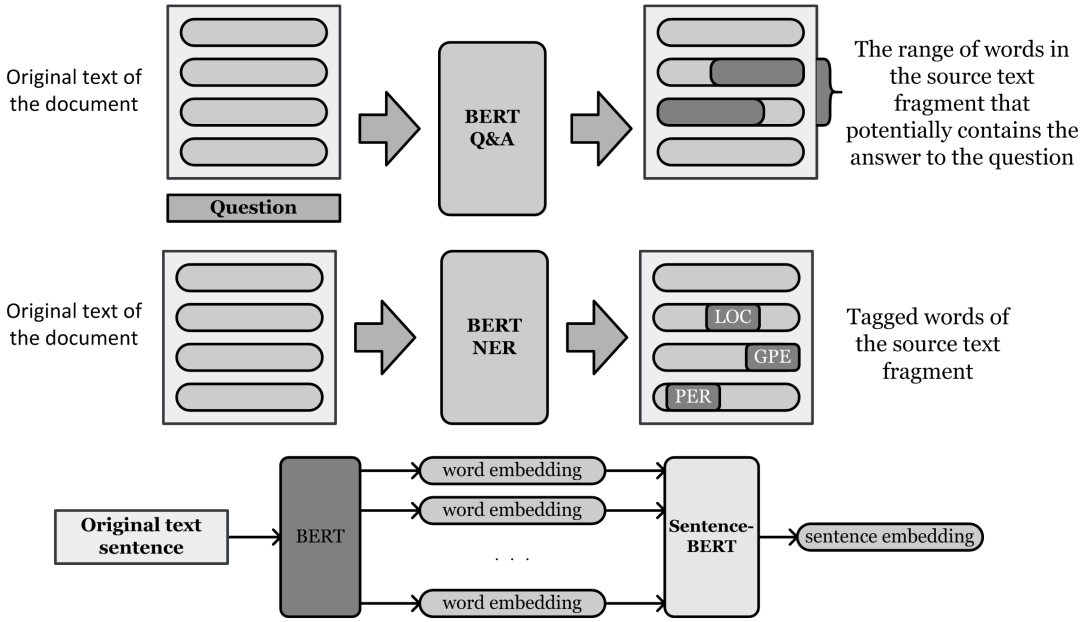


Fig. 1. The concept of application of the used BERT models

models not tailored for any specific domain, whereupon they can be used to process documents of various types and topics. As a neural network framework PyTorch framework is used [27].

The concept of application of the used BERT models is shown in Fig. 1.

The BERT Q&A model is used to search for pieces of the text in the document that contain information about certain aspects of the structural model. Since the structural model is presented as a card with fields, the task of finding information about an aspect of the model becomes the task of filling in the card field dedicated to this aspect. In this case, the desired content of the card field can be searched as an answer to some given question that describes the aspect.

As an input sequence, the BERT Q&A model uses the question text and the text to search for the answer in token format. Due to the specific architecture of the BERT Q&A model, the size of the input sequence of tokens should not exceed 512 elements. This imposes restrictions on the length of an input text to search for the answer. Since the model cannot scan the entire text of the document in search of the answer at once, it is necessary to break the text into fragments of an appropriate length. In order to prepare the fragments, the text is preliminarily cleared of formatting characters and divided into sentences. The text fragment is formed by sequentially adding consecutive sentences until the maximum possible number of tokens is reached.

Sequential forming of the fragments leads to the following problem. A potential answer to the question can be voluminous, consisting of multiple sentences. In this case, the beginning of a potential answer may be in one fragment and the ending in the other. There is a risk that the BERT Q&A model will be able to identify a potential answer from the first fragment, but not from the other one. In order to reduce the risk of receiving an incomplete answer by the system, it was decided to allocate text fragments with an overlap on each other.

Not all of the formed fragments actually contain answers to the question. Sometimes the desired answer (for example, the name of the author of the document) appears in the text only 1 or 2 times. In this case, it can be present in just one text fragment. In order to reduce the number of text fragments to search for the answer, we filter the fragments using relevant words that describe the aspect. Further analysis will only consider relevant fragments (that is, fragments that contain the relevant words).

The questions that are used to prepare the input of the Q&A model represent the meaning of the aspect. Using a set of questions instead of just one question can increase the system's chances to correctly identify the data we are looking for. Thus, by dividing the text into fragments for each question from the set, filtering out the relevant ones, using the BERT Q&A model and combining the results obtained for different questions, it is possible to get an array of answers that contain information about the aspect.

To make the method adjustable for various domains, the user should be able to define and customize all of the fields of the document card. This imposes the need to provide a universal and flexible approach to setting up a field content search, which implies choosing relevant word sets and questions. However, words in the set should not be too general (in this case, filtering may be useless) or only specialized (then there may not be relevant text fragments at all). Questions should be formulated based on the following principles. First, they should be as short as possible (the size of text fragments depends on the length of the question). Second, they should be specific. Empirically, it was found that the best result was obtained using questions *Who*, *What*, *When*, *Where*. Third, all questions related to the same field should receive the same answer according to the proposed methodology.

In some cases, we know exactly which type of named entities should be contained in the answers. For example, when searching for the name of the author of a document, the answers must explicitly contain an entity of the PER (person's name) category. For additional filtering (in order to remove the answers that do not contain a desired entity), it is proposed to use the BERT NER model. In order to apply this filtering, it is necessary to determine a list of required named entities for every aspect.

Document card fields can be either short (for example, document title, author's name) or voluminous (for example, a list of modern technologies mentioned in the document). In this regard, it is necessary to use different strategies for processing the received array of answers in order to form the final content of the card field.

Taking into account the chosen methodology for using the BERT Q&A model and BERT NER model to form the content of card fields, each field (that is, each aspect of the structural model) should be provided with the following data:

- a set of questions, the answers to which should be included in the content of a field;
- a set of words which are thematically relevant to the content of the card field;
- a set of categories of named entities (named entities tags);
- a field type mark (short or voluminous).

By determining the set of fields and providing each of them with such data, the proposed method can be adjusted to different types of documents and various topics.

The full scheme of the proposed concept of using the BERT models to form the content of the document card field is shown in Fig. 2.

On Stage 1 the data is prepared for processing. The source text of the document, extracted from the PDF file, is cleaned of formatting characters and split into sentences.

Stage 2 consists of breaking the source text into text fragments for search and forming input sequences for the BERT Q&A model. The generated fragments are filtered using the relevant words. If the fragment does not meet the minimum requirements of relevant words, it is discarded and cannot be used for further analysis.

At Stage 3 the BERT Q&A model is applied to those text fragments that contain the relevant words. As a result, at Stage 4, an array of answers is received.

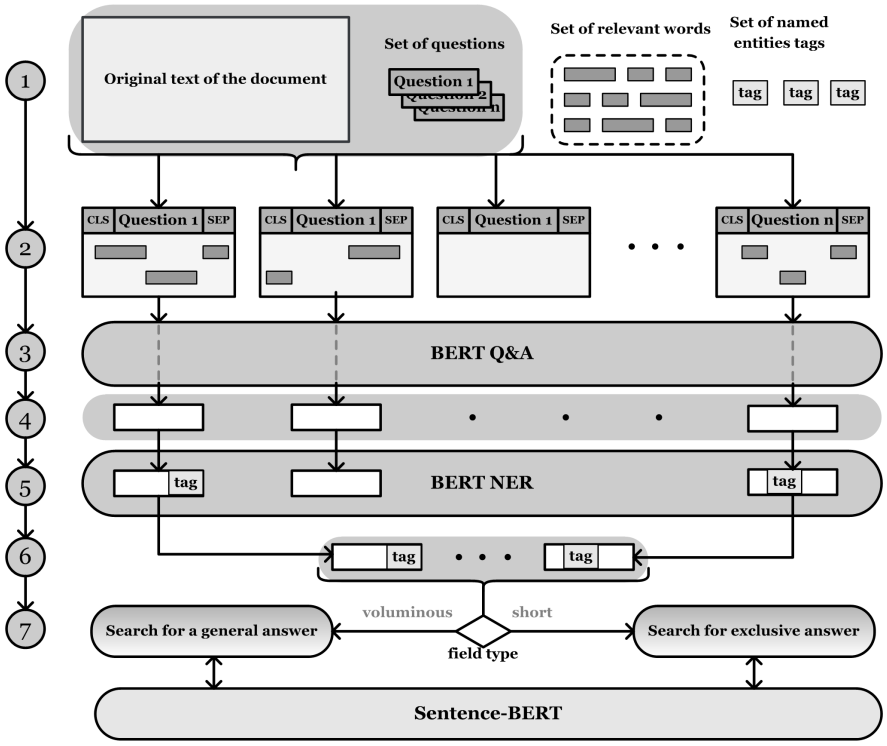


Fig. 2. The proposed concept of using the BERT models to form the content of the document card field

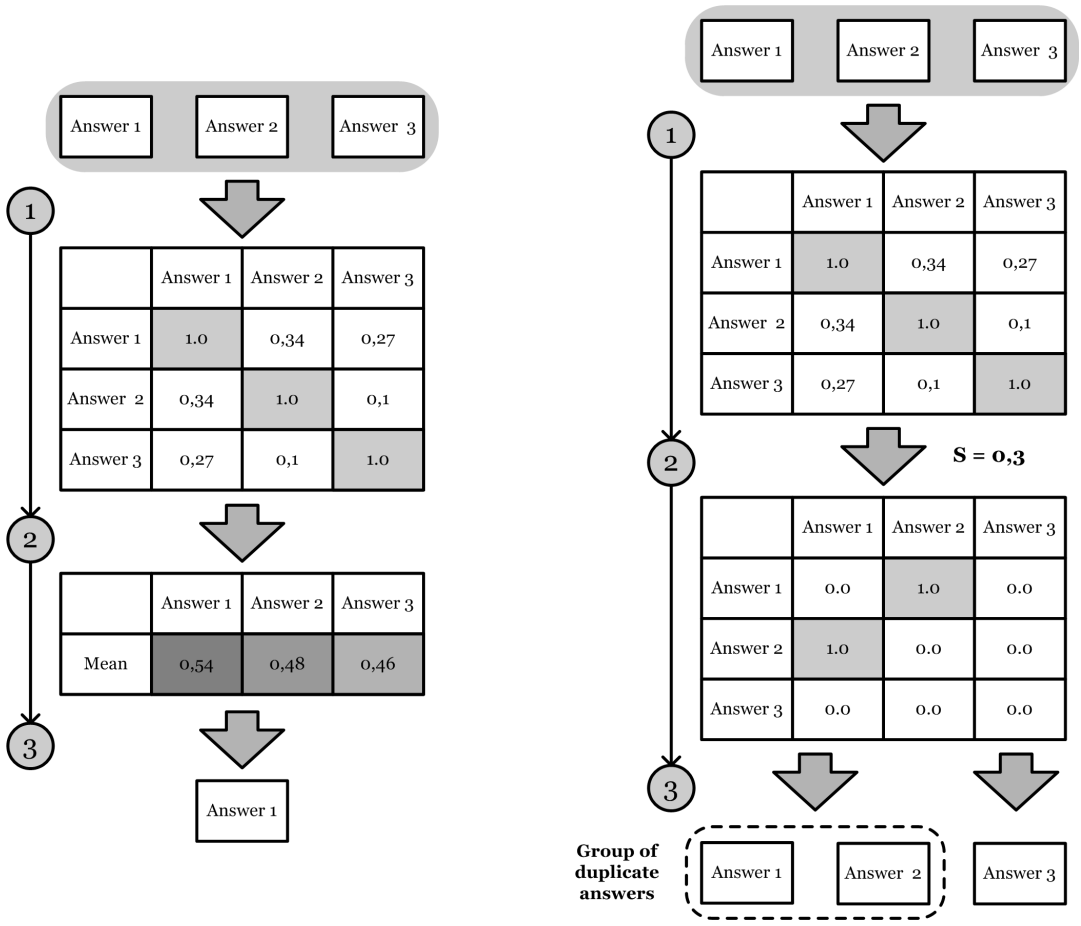
At Stage 5, the answers from the received array are fed to the BERT NER model in order to search them for named entities of the specified categories. Answers in which entities are not found are filtered out. As a result, at Stage 6, an array of answers containing the words of the required categories is obtained.

Stage 7 consists of the formation of the final content of the card field. One of two algorithms is applied to the array of answers, depending on the type of the field being processed. In the case of a short field, an exclusive answer choosing algorithm is applied, and in the case of a voluminous field, a generalizing answer forming algorithm is applied. The visualization of the proposed algorithms is shown in Fig. 3.

The algorithm for choosing an exclusive answer (Fig. 3a) is implemented as follows. A semantic similarity matrix is formed for the array of answers, where each cell of the matrix contains the result of comparing two corresponding answers. The comparison is performed by calculating the cosine measure of similarity between the vector representations of the answers that are obtained via Sentence-BERT model. For each of the answers, the average value of its measure of similarity with other answers in the array is calculated. The answer with the highest average value is chosen as an exclusive answer and used to fill the content of the card field.

The generalizing answer forming algorithm includes the search for duplicate answers in the array (Fig. 3b). For answers, a similarity matrix is constructed in a similar way. Then, the numerical values in the cells are converted according to the rule:

$$Sim'_{ij} = \begin{cases} 0, & \text{if } Sim_{ij} < S \text{ or } i = j, \\ 1, & \text{if } Sim_{ij} > S. \end{cases} \tag{1}$$



(a) Exclusive answer choosing algorithm (b) Generalizing answer forming algorithm

Fig. 3. The visualization of the proposed algorithms

Here Sim'_{ij} is the new content of the cell, Sim_{ij} is the initial value, S is the similarity threshold value set by the user.

Cells containing the ones are grouped, resulting in clusters of duplicate answers. From each cluster, the best answer is chosen using the algorithm for choosing an exclusive answer that has been described earlier. After that, for the final array of unique answers, which are words or phrases, the original full sentences are extracted from the original text of the document. The concatenation of these sentences is considered as the generalizing answer and used to fill the content of the card field.

4. Software Implementation of the Proposed Method

In order to test the proposed method we implemented a system for building structural models of text documents. The system consists of three main blocks shown in Fig. 4: the interface block, the data processing block and the data storage block.

The interface block serves as the means for the user to interact with the system. Through this block, the user can upload files for further processing, launch, manage and monitor data processing, and also view the constructed structural models of documents. The block is implemented as a Docker container with the apache2 web server running in it, which processes the requested PHP pages.

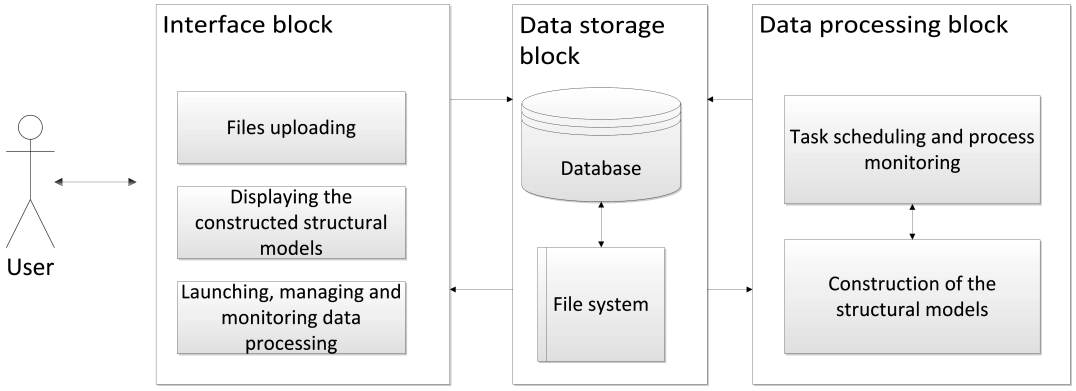


Fig. 4. The block scheme of the system for building structural models of text documents

The data processing block is the core of the system. This block is also implemented as a separate Docker container with a task scheduler running in it, which periodically checks the database for new tasks and starts their execution, in particular, the construction of structural models of documents. Task scheduling and data processing functions are implemented as Python scripts.

The data storage block serves as a link between the interface block and the data processing block. Documents uploaded by the user enter the file system, and information about the uploaded files is recorded in the database. The task scheduler uses the database to store data about scheduled and running processes. Structural models constructed by the data processing block are also stored in the database.

5. Experimental Results

We conducted a series of experiments using several sets of documents. The first set was compiled from scientific articles related to artificial intelligence and data processing that were presented in recent years at ACM conferences such as ICMLC (machine learning and computing), IR (information retrieval) and WSDM (web search and data mining). The second set was compiled from archives of AIAA Journal of Aircraft and AIAA Journal of Spacecraft and Rockets (about 16,000 articles in total). The articles from the sets were processed using the system and structural models were generated for each of them.

Before processing the articles, we set up the system by preparing the initial data for 9 fields: 4 short and 5 voluminous. Short fields described the general features of the article (title and authors) and the conference where it was presented (name and date). Voluminous fields described various aspects of the article's content such as the goal and relevance of the research, existing and proposed models and methods, performance and quality of the proposed methods.

In order to assess the quality of the proposed method, we prepared reference structured models for the articles by manually extracting fragments of the articles that described the desired aspects. Then, models generated by the system were compared to models manually prepared by experts. As the result of the models comparison, the degree of semantic similarity was calculated for each field. The similarity score takes values from 0 to 1 and shows how well the field value generated by the system corresponds to the value determined by the expert. The similarity score for short fields is determined using a Sentence BERT model and a measure of cosine similarity. The similarity score for voluminous fields is calculated via a modified Jaccard's binary similarity

measure as the percentage of similar sentences in the compared texts relative to the total number of unique sentences in both texts.

Table 1 demonstrates the result of comparison of models generated by the system and manually prepared by experts. By "good match" we denote a case in which the card field generated by the system allows to largely learn the respective aspect of the article without reading the article itself (which empirically corresponds to the similarity score exceeding 0.7). By "partial match" we denote a case where at least part of the information related to the aspect is present in the card field generated by the system (which corresponds to the similarity score exceeding 0.3). The last column of the table ("At least partial matches") contains the fraction of articles for which the content of a card field generated by the system fully or partially matches the content prepared by an expert. We use this value to assess the quality of the structured models built by the system.

Table 1. The result of comparison of models generated by the system and prepared by experts

Field name	Type	Good matches	Partial matches	At least partial matches
Title of the article	short	21%	31%	52%
Authors of the article	short	47%	0	47%
Conference name	short	68%	0	68%
Conference date	short	68%	0	68%
The goal of the research	voluminous	31%	21%	52%
Relevance of the research	voluminous	37%	37%	74%
Existing models/methods	voluminous	26%	37%	63%
Proposed models/methods	voluminous	42%	32%	74%
Performance and quality	voluminous	53%	21%	74%

The experiment demonstrated relatively high quality for voluminous fields. The "Relevance of the research", "Proposed models and methods" and "Performance and quality" fields were at least partially correctly recognized for 74% of articles, the "Existing models and methods" field — for 63% of articles. Further improvement of quality may be achieved by adjusting sets of questions and words for aspects. In general, the results of the experiment demonstrated the ability of the system to extract fragments of interest from various unstructured natural language texts in a uniform manner.

Aspects that had the lowest quality of recognition were: "Title of the article" and "Authors of the article" (52% and 47% respectively). This is because the proposed method extracts information based on the proximity of the meaning of the extracted fragment and its context to the question. However, the title and the list of authors have no context: they are located separately in a special place of the article. Also, the content of these fields is unique for each article and cannot be described by a set of questions. Therefore, to extract these aspects, it is necessary to use other methods based on utilizing information about the structure of the article.

In order to check the versatility of the developed method, we also carried out experiments using another type of documents, namely national strategies in technical areas such as Artificial Intelligence. Experiments showed the ability of the system to extract aspects such as goals of the strategy, organizations responsible for implementation of the strategy and technologies that are used to achieve the strategy's goals.

The experiments were carried out on sets of documents of various sizes (from tens to tens of thousands of documents). They confirmed the scalability of the system.

Conclusion

In this work, we proposed an Information Extraction method that forms structural models of text documents. A structural model of a document is a card, each field of which contains text that describes a certain aspect of the document. The proposed method is based on applying a Question Answering neural network model to fragments of the text in the document in order to generate answers that potentially describe the target aspect. The fragments are filtered using sets of relevant words and lists of required named entities. The array of answers generated by the Q&A model is used to form a card field of a document. The formation of the field is performed using one of two algorithms depending on the type of the field. In the case of a short field, an exclusive answer choosing algorithm is applied, and in the case of a voluminous field, a generalizing answer forming algorithm is applied.

We presented a system for building structural models of documents that implements the proposed method. The system allows the user to upload documents, manage their processing and view the constructed structural models. We analyzed the quality of the proposed method via an experiment conducted on a set of scientific articles.

We plan to further develop the proposed method in order to overcome the discovered problems and increase the quality of recognition. We plan to conduct additional experiments in order to evaluate the contribution of using NER to the quality of information extraction, and also to compare the proposed method with other approaches such as rule-based and probabilistic approach.

This paper is a part of the research work carried out within the Bauman Deep Analytics project of the Priority 2030 program.

References

1. Mansouri A., Affendey L.S., Mamat A. Named entity recognition approaches. International Journal of Computer Science and Network Security. 2008. Vol. 8, no. 2. P. 339–344.

2. Brown D.E., Liu X. Extracting Addresses from News Reports Using Conditional Random Fields. Proceedings of the 15th IEEE International Conference on Machine Learning and Applications, ICMLA, Anaheim, California, USA, December 18–20, 2016. IEEE, 2016. P. 791–795. DOI: 10.1109/ICMLA.2016.0141.

3. Benson E., Haghighi A., Barzilay R. Event discovery in social media feeds. Association for Computational Linguistics: Human Language Technologies, 49th Annual Meeting, HLT '11, Portland, Oregon, USA, June 19–24, 2011. Proceedings. Vol. 1. Association for Computational Linguistics, 2011. P. 389–398.

4. Turmo J., Ageno A., Catala N. Adaptive information extraction. ACM Computing Surveys. 2006. Vol. 38, no. 2. P. 1–47. DOI: 10.1145/1132956/1132957.

5. Chai J.Y., Biermann A.W., Guinn C.I. Two dimensional generalization in information extraction. Proceedings of the Sixteenth National Conference on Artificial Intelligence, AAAI-99, Orlando, Florida, USA, July 18–22, 1999. American Association for Artificial Intelligence, 1999. P. 431–438.

6. García-Constantino M., Atkinson K., Bollegala D., *et al.* CLIEL: Context-based information extraction from commercial law documents. Proceedings of the 16th International Conference on Artificial Intelligence and Law, ICAIL'17, London, UK, June 12–16, 2017. Association for Computing Machinery, 2017. P. 79–87. DOI: 10.1145/3086512.3086520.

7. Kadhim K.J., Sadiq A.T., Abdulah H.S. Unsupervised-Based Information Extraction from Unstructured Arabic Legal Documents. Opción: Revista de Ciencias Humanas y Sociales. 2019. Vol. 35, no. 20. P. 1097–1117.

8. Freitag D. Machine learning for information extraction in informal domains. Machine learning. 2000. Vol. 39, no. 2. P. 169–202. DOI: 10.1023/A:1007601113994.

9. Borkar V., Deshmukh K., Sarawagi S. Automatic segmentation of text into structured records. Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data, SIGMOD'01, Santa Barbara, California, USA, May 21–24, 2001. Association for Computing Machinery, 2001. P. 175–186. DOI: 10.1145/375663.375682.

10. McCallum A. Efficiently inducing features of conditional random fields. Uncertainty in Artificial Intelligence, Proceedings of the Nineteenth Conference, UAI03, Acapulco, Mexico, August 7–10, 2003. Morgan Kaufmann, 2003. P. 403–410.

11. Feldman R., Sanger J. Probabilistic Models for Information Extraction. The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, 2006. P. 131–145.

12. Wang A., Singh A., Michael J., *et al.* GLUE: a multi-task benchmark and analysis platform for natural language understanding. Proceedings of the 7th International Conference on Learning Representations, ICLR 2019, New Orleans, Louisiana, USA, May 6–9, 2019. P. 1–20. DOI: 10.18653/v1/w18-5446.

13. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, Minnesota, USA, June 2–7, 2019. Vol. 1: Long and Short Papers. Association for Computational Linguistics, 2019. P. 4171–4186. DOI: 10.18653/v1/n19-1423.

14. Pennington J., Socher R., Manning C.D. Glove: Global vectors for word representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, Doha, Qatar, October 25–29, 2014. Association for Computational Linguistics, 2014. P. 1532–1543. DOI: 10.3115/v1/d14-1162.

15. Burtsev M., Seliverstov A., Airapetyan R., *et al.* DeepPavlov: Open-Source Library for Dialogue Systems. Association for Computational Linguistics-System Demonstrations, Proceedings of the 56th Annual Meeting, Melbourne, Australia, July 15–20, 2018. Association for Computational Linguistics, 2018. P. 122–127. DOI: 10.18653/v1/p18-4021.

16. Xue K., Zhou Y., Ma Z., *et al.* Fine-tuning BERT for joint entity and relation extraction in Chinese medical text. Proceedings of the 2019 IEEE International Conference on Bioinformatics and Biomedicine, BIBM, San Diego, California, USA, November 18–21, 2019. IEEE, 2019. P. 892–897. DOI: 10.1109/bibm47256.2019.8983370.

17. Wang Q., Yang L., Kanagal B., *et al.* Learning to Extract Attribute Value from Product via Question Answering: A Multi-task Approach. Proceedings of the 26th ACM

SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD'20, USA, August 23–27, 2020. Association for Computing Machinery, 2020. P. 47–55. DOI: 10.1145/3394486.3403047.

18. Banerjee P., Pal K.K., Devarakonda M.V., Baral C. Biomedical Named Entity Recognition via Knowledge Guidance and Question Answering. ACM Transactions on Computing for Healthcare. 2021. Vol. 2, no. 4. P. 1–24. DOI: 10.1145/3465221.
19. Li X., Yin F., Sun Z., *et al.* Entity-Relation Extraction as Multi-Turn Question Answering. Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 – August 2, 2019. Vol. 1: Long Papers. Association for Computational Linguistics, 2019. P. 1340–1350. DOI: 10.18653/v1/p19-1129.
20. Qiu L., Ru D., Long Q., Zhang W., Yu Y. QA4IE: A Question Answering Based Framework for Information Extraction. Proceedings of the 17th International Semantic Web Conference, ISWC 2018, Monterey, California, USA, October 8–12, 2018. Vol. 11136 / ed. by D. Vrandečić, K. Bontcheva, M.C. Suárez-Figueroa, *et al.* Springer, 2018. P. 198–216. Lecture Notes in Computer Science. DOI: 10.1007/978-3-030-00671-6_12.
21. Rajpurkar P., Jia R., Liang P. Know What You Don't Know: Unanswerable Questions for SQuAD. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15–20, 2018. Vol. 2: Short Papers. Association for Computational Linguistics, 2018. P. 784–789. DOI: 10.18653/v1/p18-2124.
22. Weischedel R., Hovy E., Marcus R., *et al.* OntoNotes: A large training corpus for enhanced processing. Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitation / ed. by J. Olive, C. Christianson, J. McCary. Springer, 2011.
23. Google Research Github Account. TensorFlow code and pre-trained models for BERT. URL: `https://github.com/google-research/bert` (accessed: 31.10.2022).
24. DeepPavlov lab Github Account. An open source library for deep learning end to end dialog systems and chatbots. URL: `https://github.com/deeppavlov/DeepPavlov` (accessed: 31.10.2022).
25. Reimers N., Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP, Hong Kong, China, November 3–7, 2019. Association for Computational Linguistics, 2019. P. 3982–3992. DOI: 10.18653/v1/D19-1410.
26. Ubiquitous Knowledge Processing Lab Github Account. Multilingual Sentence & Image Embeddings with BERT. URL: `https://github.com/UKPLab/sentence-transformers` (accessed: 31.10.2022).
27. An open source machine learning framework PyTorch. URL: `https://pytorch.org/` (accessed: 31.10.2022).

УДК 004.89 DOI: 10.14529/cmse230102

МЕТОД СОЗДАНИЯ СТРУКТУРНЫХ МОДЕЛЕЙ ТЕКСТОВЫХ ДОКУМЕНТОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

© 2023 Д.В. Березкин, И.А. Козлов, П.А. Мартынюк, А.М. Панфилкин

Московский государственный технический университет имени Н.Э. Баумана

(105005 Москва, ул. 2-я Бауманская, д. 5, стр. 1)

E-mail: berezkind@bmstu.ru, kozlovilya89@gmail.com,

martapauline@yandex.ru, panfilkinam@student.bmstu.ru

Поступила в редакцию: 03.11.2022

В статье описываются современные нейросетевые модели на основе BERT и рассматривается их применение для задач обработки естественного языка (NLP), таких как ответы на вопросы и распознавание именованных сущностей. В статье представлен метод решения задачи автоматического создания структурных моделей текстовых документов. Предлагаемый метод является гибридным и основан на совместном использовании нескольких моделей NLP. Метод строит структурную модель документа, извлекая предложения, соответствующие различным аспектам документа. Извлечение информации осуществляется с использованием вопросно-ответной модели BERT с вопросами, подготовленными отдельно для каждого аспекта. Ответы фильтруются с помощью модели распознавания именованных сущностей BERT и используются для формирования содержимого каждого поля структурной модели. В статье предложены два алгоритма формирования содержимого поля — алгоритм выбора исключающего ответа и алгоритм формирования обобщающего ответа, которые используются для коротких и объемных полей соответственно. В статье также описывается программная реализация предлагаемого метода и обсуждаются результаты экспериментов, проведенных для оценки качества метода.

Ключевые слова: извлечение информации, нейронная сеть, распознавание именованных сущностей, вопросно-ответная система.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Berezkin D.V., Kozlov I.A., Martynyuk P.A., Panfilkin A.M. A Method for Creating Structural Models of Text Documents Using Neural Networks // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2023. Т. 12, № 1. С. 28–45. DOI: 10.14529/cmse230102.

Литература

1. Mansouri A., Affendey L.S., Mamat A. Named entity recognition approaches // International Journal of Computer Science and Network Security. 2008. Vol. 8, no. 2. P. 339–344

2. Brown D.E., Liu X. Extracting Addresses from News Reports Using Conditional Random Fields // Proceedings of the 15th IEEE International Conference on Machine Learning and Applications, ICMLA, Anaheim, California, USA, December 18–20, 2016. IEEE, 2016. P. 791–795. DOI: 10.1109/ICMLA.2016.0141.

3. Benson E., Haghighi A., Barzilay R. Event discovery in social media feeds // Association for Computational Linguistics: Human Language Technologies, 49th Annual Meeting, HLT '11, Portland, Oregon, USA, June 19–24, 2011. Proceedings. Vol. 1. Association for Computational Linguistics, 2011. P. 389–398.

4. Turmo J., Ageno A., Catala N. Adaptive information extraction // ACM Computing Surveys. 2006. Vol. 38, no. 2. P. 1–47. DOI: 10.1145/1132956/1132957.

5. Chai J.Y., Biermann A.W., Guinn C.I. Two dimensional generalization in information extraction // Proceedings of the Sixteenth National Conference on Artificial Intelligence, AAAI-99, Orlando, Florida, USA, July 18–22, 1999. American Association for Artificial Intelligence, 1999. P. 431–438.

6. García-Constantino M., Atkinson K., Bollegala D., *et al.* CLIEL: Context-based information extraction from commercial law documents // Proceedings of the 16th International Conference on Artificial Intelligence and Law, ICAIL'17, London, UK, June 12–16, 2017. Association for Computing Machinery, 2017. P. 79–87. DOI: 10.1145/3086512.3086520.

7. Kadhim K.J., Sadiq A.T., Abdulah H.S. Unsupervised-Based Information Extraction from Unstructured Arabic Legal Documents // Opción: Revista de Ciencias Humanas y Sociales. 2019. Vol. 35, no. 20. P. 1097–1117.

8. Freitag D. Machine learning for information extraction in informal domains // Machine learning. 2000. Vol. 39, no. 2. P. 169–202. DOI: 10.1023/A:1007601113994.

9. Borkar V., Deshmukh K., Sarawagi S. Automatic segmentation of text into structured records // Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data, SIGMOD'01, Santa Barbara, California, USA, May 21–24, 2001. Association for Computing Machinery, 2001. P. 175–186. DOI: 10.1145/375663.375682.

10. McCallum A. Efficiently inducing features of conditional random fields // Uncertainty in Artificial Intelligence, Proceedings of the Nineteenth Conference, UAI03, Acapulco, Mexico, August 7–10, 2003. Morgan Kaufmann, 2003. P. 403–410.

11. Feldman R., Sanger J. Probabilistic Models for Information Extraction // The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, 2006. P. 131–145.

12. Wang A., Singh A., Michael J., *et al.* GLUE: a multi-task benchmark and analysis platform for natural language understanding // Proceedings of the 7th International Conference on Learning Representations, ICLR 2019, New Orleans, Louisiana, USA, May 6–9, 2019. P. 1–20. DOI: 10.18653/v1/w18-5446.

13. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, Minnesota, USA, June 2–7, 2019. Vol. 1: Long and Short Papers. Association for Computational Linguistics, 2019. P. 4171–4186. DOI: 10.18653/v1/n19-1423.

14. Pennington J., Socher R., Manning C.D. Glove: Global vectors for word representation // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, Doha, Qatar, October 25–29, 2014. Association for Computational Linguistics, 2014. P. 1532–1543. DOI: 10.3115/v1/d14-1162.

15. Burtsev M., Seliverstov A., Airapetyan R., *et al.* DeepPavlov: Open-Source Library for Dialogue Systems // Association for Computational Linguistics-System Demonstrations, Proceedings of the 56th Annual Meeting, Melbourne, Australia, July 15–20, 2018. Association for Computational Linguistics, 2018. P. 122–127. DOI: 10.18653/v1/p18-4021.

16. Xue K., Zhou Y., Ma Z., *et al.* Fine-tuning BERT for joint entity and relation extraction in Chinese medical text // Proceedings of the 2019 IEEE International Conference on Bioinformatics and Biomedicine, BIBM, San Diego, California, USA, November 18–21, 2019. IEEE, 2019. P. 892–897. DOI: 10.1109/bibm47256.2019.8983370.

17. Wang Q., Yang L., Kanagal B., *et al.* Learning to Extract Attribute Value from Product via Question Answering: A Multi-task Approach // Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD'20, USA, August 23–27, 2020. Association for Computing Machinery, 2020. P. 47–55. DOI: 10.1145/3394486.3403047.

18. Banerjee P., Pal K.K., Devarakonda M.V., Baral C. Biomedical Named Entity Recognition via Knowledge Guidance and Question Answering // ACM Transactions on Computing for Healthcare. 2021. Vol. 2, no. 4. P. 1–24. DOI: 10.1145/3465221.

19. Li X., Yin F., Sun Z., *et al.* Entity-Relation Extraction as Multi-Turn Question Answering // Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 – August 2, 2019. Vol. 1: Long Papers. Association for Computational Linguistics, 2019. P. 1340–1350. DOI: 10.18653/v1/p19-1129.

20. Qiu L., Ru D., Long Q., *et al.* QA4IE: A Question Answering Based Framework for Information Extraction // Proceedings of the 17th International Semantic Web Conference, ISWC 2018, Monterey, California, USA, October 8–12, 2018. Vol. 11136 / ed. by D. Vrandečić, K. Bontcheva, M.C. Suárez-Figueroa, *et al.* Springer, 2018. P. 198–216. Lecture Notes in Computer Science. DOI: 10.1007/978-3-030-00671-6_12.

21. Rajpurkar P., Jia R., Liang P. Know What You Don't Know: Unanswerable Questions for SQuAD // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15–20, 2018. Vol. 2: Short Papers. Association for Computational Linguistics, 2018. P. 784–789. DOI: 10.18653/v1/p18-2124.

22. Weischedel R., Hovy E., Marcus R., *et al.* OntoNotes: A large training corpus for enhanced processing // Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitation / ed. by J. Olive, C. Christianson, J. McCary. Springer, 2011.

23. Google Research Github Account. TensorFlow code and pre-trained models for BERT. URL: `https://github.com/google-research/bert` (дата обращения: 31.10.2022).

24. DeepPavlov lab Github Account. An open source library for deep learning end to end dialog systems and chatbots. URL: `https://github.com/deeppavlov/DeepPavlov` (дата обращения: 31.10.2022).

25. Reimers N., Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP, Hong Kong, China, November 3–7, 2019. Association for Computational Linguistics, 2019. P. 3982–3992. DOI: 10.18653/v1/D19-1410.

26. Ubiquitous Knowledge Processing Lab Github Account. Multilingual Sentence & Image Embeddings with BERT. URL: `https://github.com/UKPLab/sentence-transformers` (дата обращения: 31.10.2022).

27. An open source machine learning framework PyTorch. URL: `https://pytorch.org/` (дата обращения: 31.10.2022).

Березкин Дмитрий Валерьевич, к.т.н., доцент, кафедра «Компьютерные системы и сети», Московский государственный технический университет имени Н.Э. Баумана (Москва, Российская Федерация)

Козлов Илья Андреевич, магистр, младший научный сотрудник, научно-учебный комплекс «Информатика и системы управления», Московский государственный технический университет имени Н.Э. Баумана (Москва, Российская Федерация)

Мартынюк Полина Антоновна, магистрант, кафедра «Компьютерные системы и сети», Московский государственный технический университет имени Н.Э. Баумана (Москва, Российская Федерация)

Панфилкин Артем Михайлович, магистрант, кафедра «Компьютерные системы и сети», Московский государственный технический университет имени Н.Э. Баумана (Москва, Российская Федерация)

УДК 004.855.5 DOI: 10.14529/cmse230103

РАСПОЗНАВАНИЕ УТОМЛЕНИЯ ЧЕЛОВЕКА НА ОСНОВЕ АНАЛИЗА ЕГО РЕЧИ С ПОМОЩЬЮ НЕЙРОСЕТЕВЫХ ТЕХНОЛОГИЙ*

© 2023 А.В. Яковлев[1,2] , В.О. Матыцин[1,3], В.А. Велюга[2], К.А Найденова[1], В.А. Пархоменко[4]

[1] *Военно-медицинская академия им. С.М. Кирова (194044 Санкт-Петербург, ул. Академика Лебедева, д. 6),*

[2] *Санкт-Петербургский государственный университет аэрокосмического приборостроения (190000 Санкт-Петербург, ул. Большая Морская, д. 67),*

[3] *Первый Санкт-Петербургский государственный медицинский университет им. И.П. Павлова Минздрава России (197022 Санкт-Петербург, ул. Льва Толстого, д. 6-8),*

[4] *Санкт-Петербургский политехнический университет Петра Великого (195251 Санкт-Петербург, ул. Политехническая, д. 29)*

E-mail: sven-7@mail.ru, matitsin@list.ru, vladislav805@yandex.com, ksennaidd@gmail.com, parhomenko.v@gmail.com

Поступила в редакцию: 15.11.2022

Качественные психофизиологические исследования сопряжены с созданием доступных и хорошо организованных баз данных, требующих большую предварительную работу по разработке измерительных комплексов, включающих не только средства для измерения психофизиологических параметров человека, но и его эмоционального состояния, которое отображается в выражении лица, речи и поведенческих паттернах респондентов. Измерительные комплексы должны также включать и средства обработки экспериментального материала. Суть исследования состояла в проведении эксперимента по созданию прототипа базы речевых данных русскоязычных респондентов, получения ответов на методические вопросы, возникающие у специалистов при использовании базы для задачи распознавания состояния утомления человека. Разработан аппаратно-программный комплекс, позволяющий синхронно регистрировать психофизиологические параметры, видеозаписи поведенческих реакций и аудиозапись речи человека. В качестве модели физического утомления использовался кардиореспираторный тест с физической нагрузкой. До прохождения и после завершении теста добровольцы зачитывали набор стандартных фонетически представительных текстов. Полученные аудиозаписи обрабатывались с помощью специализированной нейронной сети, способной анализировать интегральные спектральные характеристики звука. Результаты эксперимента показали возможность распознавания состояния утомления человека по его речи, что позволяет перейти к созданию большого банка аудиозаписей и совершенствованию алгоритмов распознавания состояния утомления.

Ключевые слова: распознавание утомления, база речевых данных, инструментальный комплекс, кардио-респираторный тест, машинное обучение, глубокая нейронная сеть.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Яковлев А.В., Матыцин В.О., Велюга В.А., Найденова К.А., Пархоменко В.А. Распознавание утомления человека на основе анализа его речи с помощью нейросетевых технологий // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2023. Т. 12, № 1. С. 46–60. DOI: 10.14529/cmse230103.

*Статья рекомендована к публикации программным комитетом Международной конференции «Data Analytics and Management in Data Intensive Domains — 2022».

Введение

Работа специалистов операторного профиля (далее — специалистов) характеризуется напряжением внимания с необходимостью его переключения, а также нервно-психическим напряжением в связи с высокой ответственностью за результаты деятельности. Высокие нагрузки ведут к развитию у таких специалистов состояния утомления, что сопряжено с угрозой пропуска значимых сигналов и немотивированного реагирования на сигналы ложные. Поэтому контроль за развитием утомления у специалистов в процессе их профессиональной деятельности является актуальной задачей. Однако решение данной задачи сопряжено с рядом проблем:

- моделирование условий труда специалиста в большинстве случаев не соответствует реальным условиям его деятельности;
- попытка провести какие-либо измерения в процессе трудовой деятельности с целью определить ее эффективность, представляет помеху работе специалиста;
- прогностическая ценность существующих математико-статистических моделей оценки работоспособности специалиста невелика в силу ограничений, накладываемых в большинстве случаев небольшими размерами изученных выборок.

В настоящее время активно развиваются методы распознавания состояния человека по речи, изображению и поведению, реализуемые с помощью систем автоматизированной оценки с применением нейронных сетей [1]. Такие системы позволяют оценивать состояния человека дистанционно, не отрывая его от привычной деятельности, например, от управления автомобилем.

Для регистрации состояния утомления человека перспективным является речевой канал. Он прост, недорог и в наименьшей степени подвержен искажениям во время регистрации, по сравнению с записью видео или физиологических показателей.

Исследовательская активность, посвященная анализу речи человека сосредоточена в нескольких основных направлениях.

Первое направление сопряжено с совершенствованием алгоритмов обработки речевого сигнала. Это направление включает совершенствование программных инструментов для визуализации речевого сигнала и расчета его характерных признаков, что реализуется в частности в таких системах как PRAAT [2, 3], ISIP [4], openSmile [5]. К совершенствованию алгоритмов обработки речевого сигнала мы относим также работу по адаптации современных алгоритмов машинного обучения для решения задач анализа речи [1].

Второе направление связано с разработкой алгоритмов и теоретических подходов к распознаванию различных состояний и патологий человека на основе анализа его речи. К этому направлению можно отнести распознавание различных эмоциональных состояний, а также отклонений от нормальных психических и физиологических состояний.

Третье направление, особенно важное в области обеспечения безопасности труда специалистов операторского профиля деятельности, включает быструю и надежную оценку отдельных состояний человека, связанных с исполнением им своих функциональных обязанностей в процессе деятельности, в том числе состояния утомления [6].

Вместе с тем, основной проблемой для исследователей является наличие качественных наборов речевых данных или баз речевых данных (далее — БРД), составляющих основу машинного обучения. Трудоемкость этой проблемы состоит том, что для каждого языка необходимо создавать свои БРД. В частности, уже разработаны базы данных, содержащие большое количество записей речи дикторов, выражающих нейтральные, положительные

либо отрицательные эмоции. Эти базы данных существуют в свободном доступе и служат в качестве источника эталонных сигналов для распознавания эмоций в голосе. Однако эти БРД содержат английскую, немецкую, итальянскую речь [7, 8], при этом русскоязычной БРД с открытым доступом пока не представлено.

В настоящее время разработано значительное число алгоритмов обработки речи, изучены характеристики голоса, выявлены речевые параметры, которые способны варьировать в зависимости от функционального и эмоционального состояния человека. Однако создание алгоритмов, позволяющих распознать утомление человека по его речи в процессе профессиональной деятельности, находится на этапе разработки прежде всего по причине трудностей моделирования состояния утомления человека. Для создания такой БРД требуются ответы на следующие методические вопросы:

1. Как моделировать утомление и что является достоверным критерием наступления состояния утомления?
2. Какие тексты должны быть использованы для чтения?
3. Какой длительности должны быть речевые фрагменты, достаточные для распознавания состояния утомления?
4. Микрофоны какого качества необходимо использовать для записи речи и какой уровень «огрубления» исходных данных допустим при обучении?

Суть настоящей работы состояла в проведении эксперимента по созданию прототипа БРД русскоязычных респондентов с целью получения ответов на вышеперечисленные вопросы. Основные усилия были направлены на реализацию законченного процесса распознавания утомления, включающего подбор текстов для чтения респондентов, разработку аппаратно-программного инструментария, проведение самого эксперимента, организацию регистрируемых данных, формирование обучающей выборки с речевыми сигналами респондентов, ее преобразование в прототип БРД и, в меньшей степени, на анализ и выбор алгоритмов распознавания речи, так как в этом вопросе возможно опираться на уже готовые и проверенные модели.

В разделе 1 рассматриваются основные методы и средства извлечения первичной информации. Далее в разделе 2 приведена краткая характеристика разработанного комплекса. Разделы 3–5 посвящены дизайну, обработке и обсуждению результатов проведения эксперимента соответственно. Основные выводы работы изложены в заключении.

1. Материалы и методы

Основу эксперимента составила синхронная регистрация речи в процессе чтения добровольцем стандартных фонетически представительных текстов. В эксперименте приняло участие 9 добровольцев (здоровые мужчины в возрасте 22–25 лет), подписавших информированное согласие. Один из добровольцев принял участие в эксперименте дважды. В ходе каждого исследования доброволец читал три стандартных текста («командный текст», «проза», «стих») до и после нагрузки. Таким образов, было получено 30 исходных аудиозаписей.

Для моделирования утомления использовали кардиореспираторный тест (КаРен) с максимальной физической нагрузкой, выполняемый добровольцами на велоэргометре Ergoline, при этом контролировали кардиореспираторные и метаболические параметры добровольцев с помощью эргоспирометрического комплекса MetaLyser (Cortex, Германия).

Для записи речи одновременно использовалось два микрофона: профессиональный миниатюрный петличный микрофон AKG C 417^{III} и высокочувствительный метрологический микрофон PCB 378A14 совместно с усилителем PCB 482C. Для аналого-цифрового преобразования звукового сигнала микрофонов использовалась внешняя двухканальная звуковая карта M-AUDIO M-Track Plus (MKII).

Для распознавания рассматриваемых состояний утомления добровольцев использовалась глубокая нейронная сеть с топологией автоэнкодера реализованная в библиотеке auDeep [9]. Автоэнкодер реализован с помощью библиотеки TensorFlow версии 1.15. Обучение глубокой нейронной сети выполнялось на графической карте NVIDIA Quadro M4000.

2. Краткая характеристика разработанного комплекса для формирования БРД

Комплекс построен по архитектуре «клиент-сервер» и состоит из нескольких элементов, объединенных в локальную компьютерную сеть, включающую базу данных, размещенную на отдельном сервере в СУБД MySQL 5.8 и содержащую все собираемые данные [10].

Для удаленного управления экспериментом и доступа к таблицам и полям базы данных использовались следующие элементы комплекса: терминал оператора для удаленного управления показом текстов и записью речи добровольца; программа для записи речи добровольца по командам оператора; программа, выполняющая по командам оператора показ слайдов с текстом на проекторе для их прочтения добровольцем.

Сформированная база данных в целом, кроме сценариев и данных о добровольцах, содержит также описания классов состояний утомления и тексты, читаемые добровольцами. Описание каждого эксперимента включает: реализуемый сценарий, идентификатор добровольца, дату проведения эксперимента, речевые файлы.

Структура разработанного web-сервиса для доступа к таблицам и полям базы данных приведена на рис. 1.

3. Моделирование состояния утомления человека

В качестве моделей утомления обычно используются модель депривации сна [11], а также модель предъявления ментальной нагрузки, например тесты на переключение внимания либо решение арифметических задач в течение 100–180 минут [12–14]. Таким образом, для моделирования умственного утомления требуется значительное время, кроме того, методы моделирования умственной нагрузки сложно стандартизировать. Поэтому в данной работе было использовано физическое утомление, моделировать которое достаточно просто, проведя тест с максимальной физической нагрузкой «до отказа».

Данная модель не требует затрат большого количества времени. Было показано, что электромиографические признаки утомления мышц при физической нагрузке соответствуют достижению человеком анаэробного порога [15]. Любая нагрузка, умственная либо физическая, вызывает явления утомления, характеризующиеся изменением функционального состояния организма человека с преобладанием процессов возбуждения либо (чаще всего) торможения в центральной нервной системе. Эти процессы оказывают влияние на свойства речи человека, поскольку речь регулируется центральной нервной системой. Таким образом, модель физического утомления на данном этапе может считаться адекватной, при этом наиболее просто воспроизводимой.

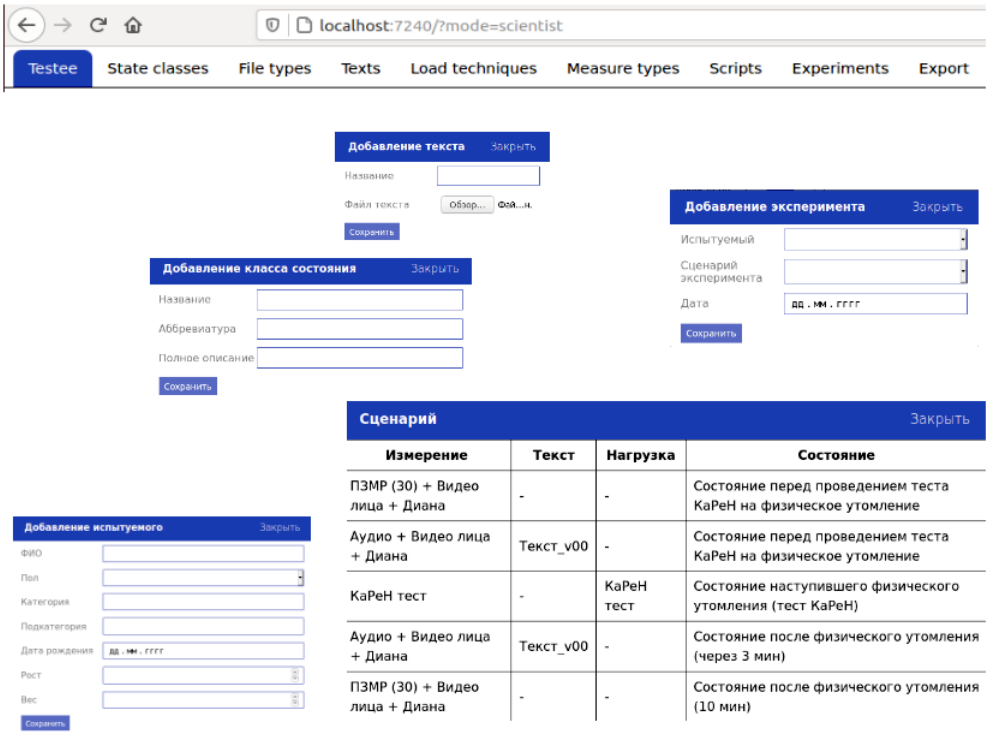


Рис. 1. Интерфейсы web-сервиса для доступа к базе данных

В качестве нагрузочного устройства в кардиореспираторном тесте (КаРен) использовали велоэргометр, поскольку он позволяет наиболее точно дозировать физическую нагрузку. Тест включает в себя фазы покоя (2 минуты), разогрева (2 минуты), нагрузки (индивидуальное время) и восстановления (1 минута).

Доброволец крутил педали со скоростью 60 об/мин. Исходная нагрузка составила 20 Вт, затем она плавно возрастала со скоростью 20 Вт/мин. Контролировалось достижение анаэробного порога (АП), критериями которого считаются следующие события: 1) вентиляция по углекислому газу VCO_2 начинает превышать вентиляцию по кислороду VO_2, соответственно значение дыхательного коэффициента (ДК) становится более 1; 2) вентиляционный эквивалент по кислороду VE/VO_2, бывший относительно постоянным, начинает расти за счет гипервентиляции; достигается точка перекреста линий регрессии VO_2 и VCO_2 за счет роста VCO_2 [16].

Доброволец выполнял работу на велоэргометре до достижения им максимально переносимой физической нагрузки, при этом достижение анаэробного порога добровольцем расценивалось как объективное подтверждение развития у него состояния физического утомления.

Исследования проводились в утренние часы, в специально оборудованном помещении. Все посторонние шумы в это время были устранены. Добровольцы приходили отдохнувшими. Перед началом каждого исследования проводилось их анкетирование с помощью анкеты САН (самочувствие, активность, настроение). В случае плохого самочувствия доброволец к исследованию не допускался.

В ходе эксперимента регистрировали два состояния добровольца: состояние перед моделированием физического утомления (состояние «не утомлен», далее — S_1) и состояние после физического утомления (через 3 минуты) (состояние «утомлен», далее — S_2). Пе-

ред предъявлением нагрузочного теста и через 3 минуты после его завершения доброволец читал специально подготовленный текст.

Выбор текста для чтения добровольцем представлял отдельную исследовательскую задачу. На момент исследования не было каких-либо достоверных сведений о том, чтение какого типа текста (командного, стихотворного и т.д.) может быть чувствительно к выявлению состояния «утомлен». Поэтому был сформирован единый текст, состоящий из небольшой тренировочной части и трех целевых частей:

– часть 1 («тренировочный текст») содержала несколько команд из ГОСТ 16600-72 для оценки средств связи (размер — 131 знак);
– часть 2 («командный текст») содержала большее количество команд из того же ГОСТ;
– часть 3 («проза») содержала фрагмент фонетически представительного текста;
– часть 4 («стих») содержала фрагмент фонетически представительного стихотворного текста.

Исходный текст автоматически разбивался на небольшие и хорошо видимые добровольцу фрагменты, которые тот читал. По мере чтения оператор комплекса давал команду на предъявление следующего фрагмента таким образом, чтобы не снижался темп чтения. При смене каждого фрагмента автоматически записывалось время относительно начала аудиозаписи. Предложенный подход хорошо себя зарекомендовал для случая чтения и дальнейшего разделения текстов разных типов внутри единого текстового документа, предъявляемого для чтения.

4. Обработка результатов исследования

Для решения задачи распознавания утомления человека была использована глубокая нейронная сеть с топологией автоэнкодера, реализованная в библиотеке auDeep [9]. Выбор указанной библиотеки определялся несколькими обстоятельствами. Во-первых, она показала высокую точность классификации акустических сцен конкурса IEEE AASP по обнаружению и классификации акустических сцен и событий (DCASE 2017) [9]. Во-вторых, она осуществляет значительный объем преобразований, связанных с корректным преобразованием исходных аудиофайлов в изображения, поступающие на вход библиотеки TensorFlow 1.15 и, соответственно, с экспортом сгенерированных признаков в формат CSV/ARFF.

Последовательность применения рассматриваемой глубокой нейронной сети для задачи распознавании утомления человека по речи представлена на рис. 2 [17].

Она состоит из шести этапов:

1. Подготовка обучающего набора данных (англ. dataset) — образцов речевых сигналов с метками классов утомления (S_1 и S_2) для «работы» с нейронной сетью. Обычно такая подготовка состоит в «оформлении» этого набора данных в соответствии с требованиями парсера, который будет «разбирать» его на этапе извлечения спектрограмм. Это один из наиболее трудоемких для исследователя этапов, так как даже небольшое отклонение при оформлении датасета от требований парсера приводит к невозможности выполнения последующих этапов.
2. Извлечение спектрограмм: извлечение спектрограмм и данных о принадлежности этих спектрограмм к рассматриваемым классам из необработанных аудиофайлов.
3. Обучение автоэнкодера на извлеченных спектрограммах.
4. Генерация признаков обученной глубокой нейронной сетью.
5. Оценка сгенерированных признаков.
6. Экспорт сгенерированных признаков в форматы CSV или ARFF.

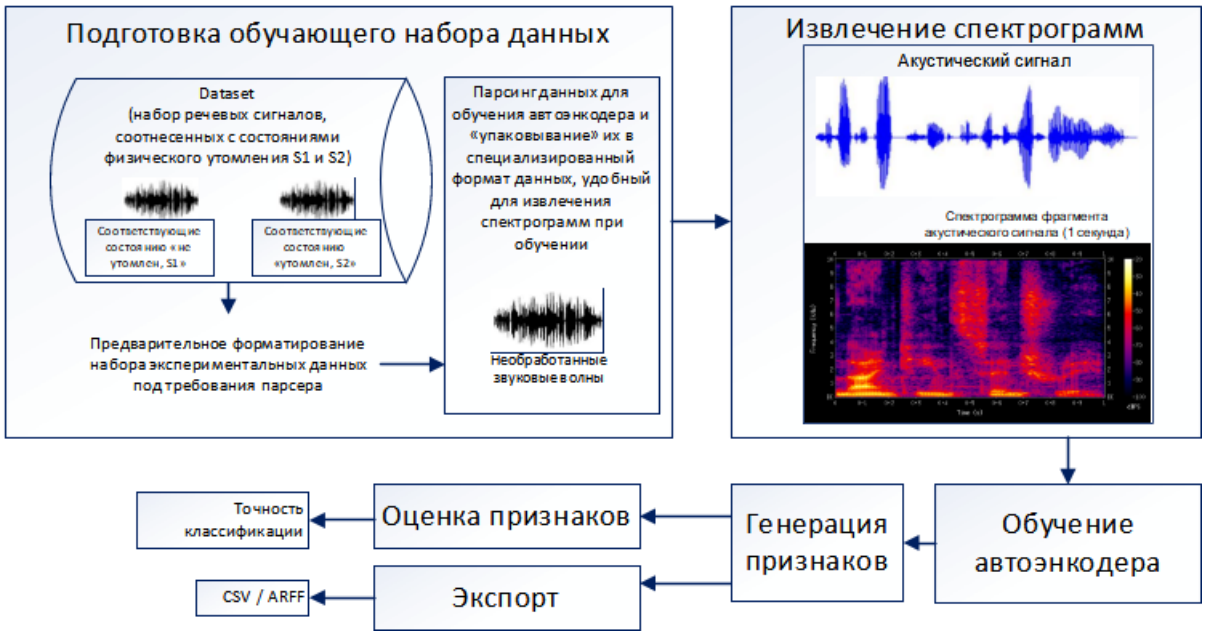


Рис. 2. Этапы использования библиотеки auDeep [9]

Для автоматизации процесса формирования датасетов в настоящей работе использовался web-сервис для доступа к базе данных (рис. 1, пункт меню «Export»). Пример результата «выгрузки» данных из БРД представлен на рис. 3.

Рис. 3. Структура выгружаемых комплексом данных

Подкаталог `data_set` содержит два подкаталога, соответствующие двум классам оцениваемого состояния утомления: подкаталог `001` содержит аудиофайлы, соответствующие состоянию добровольца «не утомлен», подкаталог `002` содержит аудиофайлы, соответствующие состоянию добровольца «утомлен». Каждый из подкаталогов содержал по 40 аудиозаписей. Таким образом, каждый из формируемых датасетов содержал 80 аудиозаписей.

В связи с тем, что целью настоящего исследования был поиск ответов на вопросы о предпочтительном типе текстов для чтения, о минимально достаточной длительности записываемых речевых фрагментов, о качестве используемых микрофонов, то варьировались датасеты, поступающие на вход нейронной сети.

Датасеты формировались исходя из значений следующих параметров:

– тип прочитанного текста, аудиозапись которого использовалась для обучения: «команда», «проза», «стихи», смешанный;
– длительность используемых для обучения нейронной сети фрагментов аудиозаписей (L): 7, 8, 9, 10 или 11 с.;
– канал регистрации: качественный петличный микрофон или высокочувствительный ненаправленный метрологический микрофон;
– уровень обрезания амплитуды записанных аудиоданных ниже указанного значения в децибелах: –45 дБ, –60 дБ, –75 дБ. Этим параметром устанавливался допустимый уровень «огрубления» исходных данных, с одной стороны, и, степень удаления из исходной записи низкоамплитудных шумов. Тем самым оценивалось, насколько используемый нейросетевой алгоритм чувствителен к шумам, присутствующих на исходной аудиозаписи.

Результаты оценки точности обучения нейронной сети — матрицы ошибок (англ. confusion matrix) для датасетов, сформированных с учетом вышеперечисленных параметров, приведены в табл. 1. Оценки качества классификации приведены в табл. 2.

Таблица 1. Матрицы ошибок обученных нейронных сетей для сформированных датасетов

Тип прочитанного текста	Уровень обрезания амплитуды аудиоданных ниже указанного значения											
	–45 дБ				–60 дБ				–75 дБ			
	Канал регистрации: качественный петличный микрофон											
	L, с.		S_1	S_2	L, с.		S_1	S_2	L, с.		S_1	S_2
Команды	7	S_1	0.70	0.30	7	S_1	0.60	0.40	7	S_1	0.64	0.36
		S_2	0.33	0.67		S_2	0.35	0.65		S_2	0.37	0.63
Проза	9	S_1	0.68	0.32	9	S_1	0.69	0.31	9	S_1	0.66	0.34
		S_2	0.27	0.73		S_2	0.21	0.79		S_2	0.32	0.68
Стихи	10	S_1	0.65	0.35	7	S_1	0.54	0.46	8	S_1	0.63	0.37
		S_2	0.26	0.74		S_2	0.37	0.63		S_2	0.35	0.65
Смешанный	9	S_1	0.80	0.20	9	S_1	0.71	0.29	8	S_1	0.68	0.32
		S_2	0.26	0.74		S_2	0.33	0.67		S_2	0.39	0.61
	Канал регистрации: высокочувствительный ненаправленный метрологический микрофон											
	L, с.		S_1	S_2	L, с.		S_1	S_2	L, с.		S_1	S_2
Команды	7	S_1	0.66	0.34	7	S_1	0.60	0.40	7	S_1	0.68	0.32
		S_2	0.33	0.67		S_2	0.31	0.69		S_2	0.44	0.56
Проза	9	S_1	0.80	0.20	9	S_1	0.79	0.21	8	S_1	0.78	0.22
		S_2	0.25	0.75		S_2	0.28	0.72		S_2	0.31	0.69
Стихи	10	S_1	0.67	0.33	10	S_1	0.56	0.44	10	S_1	0.52	0.48
		S_2	0.31	0.69		S_2	0.33	0.67		S_2	0.33	0.67
Смешанный	9	S_1	0.73	0.27	8	S_1	0.83	0.17	8	S_1	0.80	0.20
		S_2	0.29	0.71		S_2	0.39	0.61		S_2	0.43	0.57

Таблица 2. Меры точности классификации обученных нейронных сетей для сформированных датасетов

Уровень обрезания амплитуды	Тип прочитанного текста	L, с.	Меры точности классификации			
			Accuracy	Sensivity	Precision	F-мера
Канал регистрации: качественный петличный микрофон						
–45 дБ	Команды	7	0.685	0.690	0.67	0.680
	Проза	9	0.705	0.695	0.73	0.712
	Стихи	10	0.695	0.678	0.74	0.708
	Смешанный	9	0.770	0.787	0.74	**0.762**
–60 дБ	Команды	7	0.625	0.619	0.65	0.634
	Проза	9	0.74	0.718	**0.79**	**0.752**
	Стихи	7	0.585	0.577	0.63	0.602
	Смешанный	9	0.69	0.697	0.67	0.683
–75 дБ	Команды	7	0.635	0.636	0.63	0.633
	Проза	9	0.67	0.666	0.68	0.673
	Стихи	8	0.64	0.637	0.65	0.643
	Смешанный	8	0.645	0.655	0.61	0.632
Канал регистрации: высокочувствительный ненаправленный метрологический микрофон						
–45 дБ	Команды	7	0.665	0.663	0.67	0.666
	Проза	9	0.775	0.789	**0.75**	**0.769**
	Стихи	10	0.680	0.676	0.69	0.683
	Смешанный	9	0.72	0.724	0.71	0.717
–60 дБ	Команды	7	0.645	0.633	0.69	0.660
	Проза	9	0.755	0.774	0.72	0.746
	Стихи	10	0.615	0.603	0.67	0.635
	Смешанный	8	0.72	0.782	0.61	0.685
–75 дБ	Команды	7	0.620	0.636	0.56	0.595
	Проза	8	0.735	0.758	0.69	0.722
	Стихи	10	0.595	0.582	0.67	0.623
	Смешанный	8	0.685	0.740	0.57	0.644

5. Обсуждение результатов

Оценки качества распознавания целевого состояния S2 («утомлен») (табл. 1, 2) показывают, что применение нейронной сети с топологией автоэнкодер позволяет достичь точности распознавания целевого состояния до 75–79%. В зависимости от процедуры записи и характеристик записываемой речевой продукции эти результаты могут варьироваться.

Значимых отличий между использованием качественного петличного микрофона и высокочувствительного ненаправленного метрологического микрофона не выявлено. Это позволяет использовать в дальнейших исследованиях широко распространенные петличные микрофоны, но с внешней звуковой картой. Отдельным вопросом остается определение допустимых диапазонов технических характеристик используемого оборудования.

В связи с тем, что исследования проводились вне акустической безэховой камеры, а в офисном помещении в условиях относительной тишины, на аудиозаписях присутствовали шумы. Данные таблицы 2 иллюстрируют, что обрезание амплитуды исходного аудиосигнала ниже уровня –45 дБ (для аудиозаписей прозы ниже уровня –60 дБ) позволяет улучшить качество распознавания.

Одним из важных результатов исследования стало определение предпочтительного типа речевой продукции, регистрация которой позволялась распознавать нейронной сети состояние утомления с более высоким качеством. Данные таблицы 2 показывают, что использование петличного микрофона при обрезании амплитуды исходного аудиосигнала ниже уровня –45 дБ (то есть в условиях небольшого зашумления) позволяет распознавать состояние утомления со значениями F, равными 0.680 при обработке произносимых команд длительностью от 7 секунд, 0.712 — прозы длительностью от 9 секунд, 0.708 — стихотворений длительностью от 10 секунд и 0.762 для смешанного текста, состоящего из команд, прозы и стихотворений длительностью от 9 секунд. При меньшей длительности, вероятно, нейросети не хватает данных для того, чтобы найти признаки для различия состояний.

Таким образом, в результате исследования показана способность разработанной модели распознавать наступление состояния физического утомления у добровольцев по их речевой продукции.

Дальнейшее совершенствование разрабатываемой методики оценки состояния утомления по речи будет продолжено в направлении разработки БРД для исследования связи речевой продукции и умственного утомления, а также совершенствование рассмотренного в настоящей работе алгоритма для распознавания состояния утомления.

Заключение

Был проведен пилотный эксперимент с целью получения ответов на ряд методических вопросов, возникающих при создании БРД по утомлению. Для создания прототипа БРД был специально разработан аппаратно-программный комплекс для проведения исследований по распознаванию умственного и физического утомления. Была разработана методика формирования у добровольцев состояния физического утомления. Проведенный эксперимент позволил получить ответы на поставленные в начале исследования методические вопросы:

1. Для моделирования физического утомления целесообразно использовать кардиореспираторный тест с максимальной физической нагрузкой, достоверным критерием наступления состояния утомления служит достижение добровольцем анаэробного порога при выполнении этого теста.
2. При формировании БРД для чтения целесообразно использовать тексты, соответствующие особенностям речевой коммуникации операторов, т.е. командные в сочетании с текстами типа «проза».
3. Регистрируемые речевые фрагменты, достаточные для распознавания состояния утомления, должны быть не менее 7–10 секунд.
4. Использование более чувствительного микрофона по сравнению с использованием петличного микрофона не дает заметной разницы качества распознавания состояния утомления.
5. Обрезание амплитуды исходного аудиосигнала ниже уровня –45 дБ (в ряде случаев ниже уровня –60 дБ) позволяет улучшить качество распознавания.

Литература

1. Zhang X.-J., Gu J.-H., Tao Z. Research of detecting fatigue from speech by PNN // 2010 International Conference on Information, Networking and Automation (ICINA). Vol. 2. 2010. P. V2278–V2281. DOI: 10.1109/ICINA.2010.5636509.
2. Krajewski J., Batliner A., Golz M. Acoustic sleepiness detection: Framework and validation of a speech-adapted pattern recognition approach // Behavior Research Methods. 2009. Vol. 41, no. 3. P. 795–804. DOI: 10.3758/BRM.41.3.795.
3. Krajewski J., Trutschel U., Golz M., *et al.* Estimating Fatigue from Predetermined Speech Samples Transmitted by Operator Communication Systems // Proceedings of the 5th International Driving Symposium on Human Factors in Driver Assessment, Training, and Vehicle Design: Driving Assessment 2009. University of Iowa, 2009. DOI: 10.17077/drivingassessment.1359.
4. Greeley H., Berg J., Friets E., *et al.* Fatigue estimation using voice analysis // Behavior Research Methods. 2007. Vol. 39, no. 3. P. 610–619. DOI: 10.3758/BF03193033.
5. openSMILE 3.0 - audEERING. Homepage. URL: https://www.audeering.com/research/opensmile/A (дата обращения: 15.11.2022).
6. Baykaner K., Huckvale M., Whiteley I., *et al.* The Prediction of Fatigue Using Speech as a Biosignal // Statistical Language and Speech Processing. Vol. 9449 / ed. by A.-H. Dediu, C. Martín-Vide, K. Vicsi. Cham: Springer, 2015. P. 8–17. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). DOI: 10.1007/978-3-319-25789-1_2.
7. Eyben F., Scherer K., Schuller B., *et al.* The Geneva Minimalistic Acoustic Parameter Set (GeMAPS) for Voice Research and Affective Computing // IEEE Transactions on Affective Computing. 2016. Vol. 7, no. 2. P. 190–202. DOI: 10.1109/TAFFC.2015.2457417.
8. Parada-Cabaleiro E., Costantini G., Batliner A., *et al.* DEMoS: an Italian emotional speech corpus: Elicitation methods, machine learning, and perception // Language Resources and Evaluation. 2020. Vol. 54, no. 2. P. 341–383. DOI: 10.1007/s10579-019-09450-y.
9. Freitag M., Amiriparian S., Pugachevskiy S., *et al.* auDeep: Unsupervised learning of representations from audio with deep recurrent neural networks // Journal of Machine Learning Research. 2018. Vol. 18. P. 1–5. URL: http://jmlr.org/papers/v18/17-406.html.
10. Яковлев А.В. Разработка распределенной программной системы для синхронизированного сбора речевых, видео- и психофизиологических данных о добровольце в процессе экспериментального исследования // Обработка, передача и защита информации в компьютерных системах '22: Сборник докладов второй международной научной конференции, Санкт-Петербург, Россия. Санкт-Петербург: Издательство Санкт-Петербургского государственного университета аэрокосмического приборостроения, 2022. С. 95–100.
11. Hidalgo-Gadea G., Kreuder A., Krajewski J., Vorstius C. Towards better microsleep predictions in fatigued drivers: exploring benefits of personality traits and IQ // Ergonomics. 2021. Vol. 64, no. 6. P. 778–792. DOI: 10.1080/00140139.2021.1882707.

12. Fan X., Zhao C., Luo H., Zhang W. An event-related potential objective evaluation study of mental fatigue based on 2-back task // Journal of biomedical engineering. 2018. Vol. 35, no. 6. P. 837–844. DOI: 10.7507/1001-5515.201801064.

13. Trejo L.J., Kochavi R., Kubitz K., *et al.* Measures and models for predicting cognitive fatigue // Biomonitoring for Physiological and Cognitive Performance during Military Operations. Vol. 5797 / ed. by J.A. Caldwell, N.J. Wesensten. International Society for Optics, Photonics. SPIE, 2005. P. 105–115. DOI: 10.1117/12.604286.

14. Yamada Y., Kobayashi M. Detecting mental fatigue from eye-tracking data gathered while watching video: Evaluation in younger and older adults // Artificial Intelligence in Medicine. 2018. Vol. 91. P. 39–48. DOI: 10.1016/j.artmed.2018.06.005.

15. Matsumoto T., Ito K., Moritani T. The relationship between anaerobic threshold and electromyographic fatigue threshold in college women // European Journal of Applied Physiology and Occupational Physiology. 1991. Vol. 63, no. 1. P. 1–5. DOI: 10.1007/BF00760792.

16. Solberg G., Robstad B., Skjønsberg O., Borchsenius F. Respiratory gas exchange indices for estimating the anaerobic threshold // Journal of Sports Science and Medicine. 2005. Vol. 4, no. 1. P. 29–36. URL: https://pubmed.ncbi.nlm.nih.gov/24431958/.

17. Яковлев А.В. Использование многослойных сетей-автоэнкодеров для распознавания усталости человека на основе речевых данных // Обработка, передача и защита информации в компьютерных системах '22: Сборник докладов второй международной научной конференции, Санкт-Петербург, Россия. Санкт-Петербург: Издательство Санкт-Петербургского государственного университета аэрокосмического приборостроения, 2022. С. 87–94.

Яковлев Александр Викторович, к.т.н., доцент, научно-исследовательский центр, Военно-медицинская академия имени С.М. Кирова (Санкт-Петербург, Российская Федерация), кафедра прикладной информатики, Санкт-Петербургский государственный университет аэрокосмического приборостроения (Санкт-Петербург, Российская Федерация)

Матыцин Вячеслав Олегович, к.м.н., научно-исследовательский центр, Военно-медицинская академия имени С.М. Кирова (Санкт-Петербург, Российская Федерация), кафедра нормальной физиологии, Первый Санкт-Петербургский государственный медицинский университет им. И.П. Павлова Минздрава России (Санкт-Петербург, Российская Федерация)

Велюга Владислав Алексеевич, студент, Санкт-Петербургский государственный университет аэрокосмического приборостроения (Санкт-Петербург, Российская Федерация)

Найденова Ксения Александровна, к.т.н., с.н.с., научно-исследовательский центр, Военно-медицинская академия имени С.М. Кирова (Санкт-Петербург, Российская Федерация)

Пархоменко Владимир Андреевич, ассистент, Высшая школа суперкомпьютерных систем и интеллектуальных технологий, Институт компьютерных наук и технологий, Санкт-Петербургский политехнический университет Петра Великого (Санкт-Петербург, Российская Федерация)

DOI: 10.14529/cmse230103

RECOGNITION OF HUMAN FATIGUE BASED ON SPEECH ANALYSIS USING NEURAL NETWORK TECHNOLOGIES

© 2023 A.V. Yakovlev[1,2], V.O. Matytsin[1,3], V.A. Velyuga[2], X.A. Naidenova[1], V.A. Parkhomenko[4]

[1]*S.M. Kirov Military Medical Academy*
(st. Akademika Lebedeva 6, St. Petersburg, 194044 Russia),
[2]*Saint-Petersburg State University of Aerospace Instrumentation*
(st. Bolshaya Morskaia 67, St. Petersburg, 190000 Russia),
[3]*Pavlov First Saint Petersburg State Medical University*
(st. Lva Tolstogo 6-8, St. Petersburg, 197022 Russia),
[4]*Peter the Great St. Petersburg Polytechnic University*
(st. Polytechnicheskaya 29, St. Petersburg, 195251 Russia)
E-mail: sven-7@mail.ru, matitsin@list.ru, vladislav805@yandex.com, ksennaidd@gmail.com, parhomenko.v@gmail.com

Received: 15.11.2022

Qualitative psychophysiological research studies are associated with the creation of accessible and well-organized databases that require a lot of preliminary work on the development of measuring complexes, including not only tools for measuring the psychophysiological parameters of a human, but also their emotional state, which is displayed in facial expression, speech and behavioral patterns. Measuring systems should also include the means of experimental material processing. The purpose of the study was to conduct an experiment on creating a prototype of the Speech Data Base of Russian-speaking respondents and to obtain answers to some methodological questions that arise among specialists when they use the database for the task of recognizing the state of human fatigue. A hardware and software complex has been developed that allows to synchronously register psychophysiological parameters, video recordings of behavioral reactions and audio recordings of human speech. As a model of physical fatigue, a cardio-respiratory test with physical activity (load) was used. Before and after completing the test, volunteers read out a set of standard phonetically representative texts. The obtained audio recordings were processed using a specialized neural network capable of analyzing the integral spectral characteristics of sound. The results of the experiment showed the possibility of recognizing the state of human fatigue based on speech analysis, which makes it possible to proceed to the creation of a large bank of audio recordings and the improvement of algorithms for recognizing the state of fatigue.

Keywords: human fatigue recognition, speech database, instrumental complex, cardio-respiratory test, machine learning, deep neural network.

FOR CITATION

Yakovlev A.V., Matytsin V.O., Velyuga V.A., Naidenova X.A., Parkhomenko V.A. Recognition of Human Fatigue Based on Speech Analysis Using Neural Network Technologies. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2023. Vol. 12, no. 1. P. 46–60. (in Russian) DOI: 10.14529/cmse230103.

References

1. Zhang X.-J., Gu J.-H., Tao Z. Research of detecting fatigue from speech by PNN. 2010 International Conference on Information, Networking and Automation (ICINA). Vol. 2. 2010. P. V2278–V2281. DOI: 10.1109/ICINA.2010.5636509.
2. Krajewski J., Batliner A., Golz M. Acoustic sleepiness detection: Framework and validation of a speech-adapted pattern recognition approach. Behavior Research Methods. 2009. Vol. 41, no. 3. P. 795–804. DOI: 10.3758/BRM.41.3.795.
3. Krajewski J., Trutschel U., Golz M., *et al.* Estimating Fatigue from Predetermined Speech Samples Transmitted by Operator Communication Systems. Proceedings of the 5th International Driving Symposium on Human Factors in Driver Assessment, Training, and Vehicle Design: Driving Assessment 2009. University of Iowa, 2009. DOI: 10.17077/drivingassessment.1359.
4. Greeley H., Berg J., Friets E., *et al.* Fatigue estimation using voice analysis. Behavior Research Methods. 2007. Vol. 39, no. 3. P. 610–619. DOI: 10.3758/BF03193033.
5. openSMILE 3.0 - audEERING. Homepage. URL: https://www.audeering.com/research/opensmile/A (accessed: 15.11.2022).
6. Baykaner K., Huckvale M., Whiteley I., *et al.* The Prediction of Fatigue Using Speech as a Biosignal. Statistical Language and Speech Processing. Vol. 9449 / ed. by A.-H. Dediu, C. Martín-Vide, K. Vicsi. Cham: Springer, 2015. P. 8–17. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). DOI: 10.1007/978-3-319-25789-1_2.
7. Eyben F., Scherer K., Schuller B., *et al.* The Geneva Minimalistic Acoustic Parameter Set (GeMAPS) for Voice Research and Affective Computing. IEEE Transactions on Affective Computing. 2016. Vol. 7, no. 2. P. 190–202. DOI: 10.1109/TAFFC.2015.2457417.
8. Parada-Cabaleiro E., Costantini G., Batliner A., *et al.* DEMoS: an Italian emotional speech corpus: Elicitation methods, machine learning, and perception. Language Resources and Evaluation. 2020. Vol. 54, no. 2. P. 341–383. DOI: 10.1007/s10579-019-09450-y.
9. Freitag M., Amiriparian S., Pugachevskiy S., *et al.* auDeep: Unsupervised learning of representations from audio with deep recurrent neural networks. Journal of Machine Learning Research. 2018. Vol. 18. P. 1–5. URL: http://jmlr.org/papers/v18/17-406.html.
10. Yakovlev A.V. Development of a distributed software system for synchronized collection of speech, video and psychophysiological data about a volunteer in the process of experimental research. Processing, Transmission and Protection of Information in Computer Systems '22: Proceedings of the Second International Scientific Conference, St. Petersburg, Russia. St. Petersburg: Publishing of the St. Petersburg State University of Aerospace Instrumentation, 2022. P. 95–100. (in Russian).
11. Hidalgo-Gadea G., Kreuder A., Krajewski J., Vorstius C. Towards better microsleep predictions in fatigued drivers: exploring benefits of personality traits and IQ. Ergonomics. 2021. Vol. 64, no. 6. P. 778–792. DOI: 10.1080/00140139.2021.1882707.
12. Fan X., Zhao C., Luo H., Zhang W. An event-related potential objective evaluation study of mental fatigue based on 2-back task. Journal of biomedical engineering. 2018. Vol. 35, no. 6. P. 837–844. DOI: 10.7507/1001-5515.201801064.

13. Trejo L.J., Kochavi R., Kubitz K., *et al.* Measures and models for predicting cognitive fatigue. Biomonitoring for Physiological and Cognitive Performance during Military Operations. Vol. 5797 / ed. by J.A. Caldwell, N.J. Wesensten. International Society for Optics, Photonics. SPIE, 2005. P. 105–115. DOI: 10.1117/12.604286.

14. Yamada Y., Kobayashi M. Detecting mental fatigue from eye-tracking data gathered while watching video: Evaluation in younger and older adults. Artificial Intelligence in Medicine. 2018. Vol. 91. P. 39–48. DOI: 10.1016/j.artmed.2018.06.005.

15. Matsumoto T., Ito K., Moritani T. The relationship between anaerobic threshold and electromyographic fatigue threshold in college women. European Journal of Applied Physiology and Occupational Physiology. 1991. Vol. 63, no. 1. P. 1–5. DOI: 10.1007/BF00760792.

16. Solberg G., Robstad B., Skjønsberg O., Borchsenius F. Respiratory gas exchange indices for estimating the anaerobic threshold. Journal of Sports Science and Medicine. 2005. Vol. 4, no. 1. P. 29–36. URL: https://pubmed.ncbi.nlm.nih.gov/24431958/.

17. Yakovlev A.V. The use of multilayer networks-autoencoders for the recognition of human fatigue on the basis of speech data. Processing, Transmission and Protection of Information in Computer Systems '22: Proceedings of the Second International Scientific Conference, St. Petersburg, Russia. St. Petersburg: Publishing of the St. Petersburg State University of Aerospace Instrumentation, 2022. P. 87–94. (in Russian).

УДК 004.8 **DOI: 10.14529/cmse230104**

ПРИМЕНЕНИЕ ТРЕТИЧНОЙ СТРУКТУРЫ АЛГЕБРАИЧЕСКОЙ БАЙЕСОВСКОЙ СЕТИ В ЗАДАЧЕ АПОСТЕРИОРНОГО ВЫВОДА

© 2023 А.А. Вяткин[1], М.В. Абрамов[1], Н.А. Харитонов[2], А.Л. Тулупьев[3]

[1] *Санкт-Петербургский федеральный исследовательский центр Российской академии наук (199178 Санкт-Петербург, 14-я лин. В.О., д. 39),*

[2] *Санкт-Петербургский государственный университет (199034 Санкт-Петербург, Университетская набережная, д. 7-9),*

[3] *Северо-Западный институт управления Российской академии народного хозяйства и государственной службы при Президенте Российской Федерации (199034 Санкт-Петербург, Средний пр. В.О., д. 57/43)*

E-mail: aav@dscs.pro, mva@dscs.pro, nak@dscs.pro, alt@dscs.pro

Поступила в редакцию: 09.12.2022

В теории алгебраических байесовских сетей существуют алгоритмы, позволяющие проводить глобальный апостериорный вывод с использованием вторичных структур. При этом построение вторичных структур предполагает использование третичной структуры. Следовательно, возникает вопрос об обособленном применении третичной структуры в задаче апостериорного вывода. Этот вопрос рассматривался ранее, но было приведено только общее описание алгоритма, при этом учитывались лишь модели со скалярными оценками вероятности истинности. В данной работе приведен алгоритм, расширяющий вышеупомянутый до возможности его использования в случае интервальных оценок. Помимо этого, важным свойством алгебраической байесовской сети является ацикличность, и корректность работы перечисленных алгоритмов обеспечивается только для ациклических сетей. Поэтому необходимо также уметь проверять ацикличность алгебраической байесовской сети с применением третичной структуры. Описание этого алгоритма также представлено в работе, в его основе лежит ранее доказанная теорема, которая связывает количество моделей фрагментов знаний в сети с количеством непустых сепараторов и количеством компонент связности сильных сужений в цикличной АБС, а также доказанная в данной статье теорема о принадлежности двух моделей фрагментов знаний к одной компоненте связности сильного сужения. Для всех разработанных алгоритмов доказана корректность работы, а также вычислена их оценка временной сложности.

Ключевые слова: алгебраические байесовские сети, фрагмент знаний, логико-вероятностный вывод, третичная структура, вероятностные графические модели, машинное обучение.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Вяткин А.А., Абрамов М.В., Харитонов Н.А., Тулупьев А.Л. Применение третичной структуры алгебраической байесовской сети в задаче апостериорного вывода // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2023. Т. 12, № 1. С. 61–88. DOI: 10.14529/cmse230104.

Введение

Вероятностные графические модели сегодня находят широкое применение в самых разных областях науки, технологий и промышленности [1]. Например, они используются в задачах распознавания и отслеживания людей на видео [2], анализа кредитного риска [3], исследования влияния человеческого фактора на морские аварии [4], оценки вероятности успеха многоходовых социоинженерных атак [5, 6]. Один из важных представителей класса вероятностных графических моделей — алгебраические байесовские сети [7]. В своей простейшей — первичной — структуре они являются набором идеалов конъюнктов со ска-

лярными или интервальными оценками вероятности истинности (набор моделей фрагментов знаний) и применяются в целях описания экспертных систем [7]. Для более широкого практического использования алгебраических байесовских сетей необходимо развить соответствующий теоретический аппарат через решение ряда задач, одной из которых посвящен данный материал.

В контексте теории алгебраических байесовских сетей рассматриваются различные вопросы, в частности, — распространение (пропагация) свидетельства, то есть пересчет оценок вероятностей истинности элементов алгебраической байесовской сети на основе свидетельства, интерпретацией которого является новая, ранее неизвестная информация о предметной области. Например, если имеется сформированная модель, описывающая прогноз погоды, то новой информацией будет утверждение о том, что завтра определенно пойдет снег. Распространение (пропагация) свидетельства является задачей глобального апостериорного вывода. Более формально, свидетельство — либо предположение о том, что какие-то утверждения оказались истинными или ложными, либо о том, что над подыдеалом, описывающем часть утверждений, задана модель фрагмента знаний с новыми оценками вероятности истинности. Для пропагации свидетельства могут использоваться вторичная или третичная структуры. На данный момент для решения этой задачи применяется механизм распространения виртуальных свидетельств во вторичной структуре алгебраической байесовской сети [8]. Вторичная структура является графом смежности, где вершины нагружены математическими моделями фрагментов знаний (идеалы конъюнктов со скалярными или интервальными оценками вероятности истинности) [8]. Виртуальным же свидетельством называют пересечение двух смежных вершин вторичной структуры, также являющееся моделью фрагмента знаний, при этом пересечение задается между парами вершин, где в первой вершине уже учтена информация, поступившая от изначального свидетельства, а во второй — нет. Вторичная структура для заданной первичной не определена однозначно, поэтому необходимо отдельно исследовать способы построения вторичных структур, в которых распространение свидетельства происходило бы наименее ресурсозатратно. Для синтеза таких вторичных структур, то есть их построения на основе первичных с, возможно, дополнительными ограничениями на вид графа, в настоящий момент используются третичные структуры [9, 10]. Третичная структура — это диаграмма Хассе над замкнутым множеством всех непустых пересечений пар моделей фрагментов знаний [11].

Иными словами, для пропагации свидетельств нужна вторичная структура, для построения которой используется третичная структура. Отсюда возникает вопрос: есть ли возможность производить апостериорный вывод, применяя только третичную структуру, без построения вторичной, что позволило бы уменьшить количество применяемых объектов и, возможно, ускорить общую работу модели? Таким образом, исследование новых алгоритмов апостериорного вывода, в частности с применением только третичной структуры, является актуальным, поскольку позволило бы достичь обозначенных выше результатов: уменьшить количество применяемых объектов и, вероятно, ускорить общую работу модели. Этот вопрос ранее рассматривался в [12], но только для скалярных оценок вероятности истинности, при этом алгоритм описывался в общих чертах, без явного представления псевдокода.

Следует также отметить, что еще важным свойством алгебраической байесовской сети является ацикличность. Ее проверка необходима, так как только для ацикличных сетей доказана корректность вышеупомянутого алгоритма, наличие такого свойства позволяет

значительно уменьшить вычислительную сложность поддержания непротиворечивости алгебраической байесовской сети, что также важно при использовании ее на практике. Таким образом, целью работы является расширение и анализ алгоритма апостериорного вывода, применяющего третичную структуру, до возможности его использования в случае интервальных оценок. В связи с этой целью выделяются следующие задачи:

1) описать расширение рассматриваемого в [12] алгоритма апостериорного вывода;
2) сформировать алгоритм проверки ацикличности алгебраической байесовской сети с применением третичной структуры, что ранее не проводилось;
3) доказать корректность работы этих алгоритмов;
4) оценить их сложность.

Теоретическая значимость работы заключается в формировании возможности дальнейшего применения полученных результатов в исследованиях задачи апостериорного вывода в алгебраических байесовских сетях, исследовании третичной структуры как самодостаточного объекта, необходимого при практическом применении алгебраических байесовских сетей. Практическая значимость заключается в возможном ускорении работы алгоритмов апостериорного вывода за счет уменьшения количества создаваемых объектов и расширении за счет этого области применения данного аппарата.

Опишем краткое содержание разделов данной статьи. В разделе 1 рассматриваются работы, в которых исследовались вопросы, связанные с изучаемыми в данной статье проблемами, а также работы, на которые опирается данная статья. Раздел 2 знакомит читателя с теоретической основой, необходимой для понимания дальнейшего изложения статьи. В разделе 3 описывается апостериорный вывод, применяющий третичную структуру. Раздел 4 посвящен алгоритму проверки ацикличности алгебраической байесовской сети. В разделе 5 кратко описываются и анализируются полученные результаты — работа алгоритмов, их сложность. Заключение подводит итоги исследования и обозначает направления дальнейших работ.

1. Релевантные работы

Помимо алгебраических байесовских сетей существуют другие классы вероятностных графических моделей, которые позволяют решать те же задачи. Наиболее распространенными являются байесовские сети доверия (БСД). БСД находят широкое применение в различных областях. Так, они используются в анализе безопасности и надежности систем, оценке рисков [13–18], обработке естественного языка [19], изучении настроения пользователей социальной сети [20], повышении безопасности сотрудников на рабочих местах [21], разработке городской инфраструктуры [22], здравоохранении [23], климатологии [24], анализе аварий при судоходстве [25], в сетях газопровода [26]. При этом в БСД используются только точечные оценки вероятности истинности, а также остаются открытыми вопросы, связанные с обработкой направленных циклов [27]. В теории АБС же могут учитываться интервальные оценки вероятности истинности, что позволяет, например, легче формализовывать неопределенность высказывания на естественном языке, учитывать наблюдения с частично пропущенными или утраченными данными [27]. Помимо этого в теории АБС существуют подходы, позволяющие использовать циклические структуры [28].

Вернемся к рассмотрению задачи построения алгоритма апостериорного вывода. Ранее использовался механизм распространения виртуальных свидетельств [8], использующий вторичную структуру, построение которой описано, например, в [29]. Данная же рабо-

та рассматривает использование только третичной структуры при проведении глобального апостериорного вывода. Фундаментом работы послужило исследование [12], в котором описывается апостериорный вывод в алгебраической байесовской сети с использованием только третичной структуры в случае скалярных оценок. Эта статья является на текущий момент единственной, посвященной подобному вопросу. В ней представлено общее описание алгоритма вывода, а также доказана его корректность для ациклических алгебраических байесовских сетей. Для применения этого алгоритма на практике требуется доработка, чтобы иметь возможность использовать его в алгебраических байесовских сетях с интервальными оценками вероятности истинности. Данная статья посвящена доработке указанного алгоритма, а также оценке его сложности, что необходимо для дальнейшего практического применения и анализа его работы.

Второй основной вопрос, которому посвящена статья — проверка ацикличности алгебраической байесовской сети. Такую проверку необходимо осуществлять, так как только для ациклических байесовских сетей доказана корректность работы вышеупомянутого алгоритма, и потому свойство ацикличности важно для корректной работы расширяющего алгоритма, описываемого в данной работе. Эта проблема рассматривалась в [30], но задача решалась с применением, опять же, новых структур (четвертичных), помимо третичных. При этом, так как предполагается использование только третичной структуры для практического применения алгебраических байесовских сетей, проверку следует осуществлять, не создавая новые структуры, то есть использовать лишь третичную структуру. Работ, посвященных задаче проверки ацикличности в такой постановке, нет. Но стоит отметить, что решение этой задачи в текущей статье опирается на теорему о цикличности первичной структуры, описанную в [31].

2. Теоретическая основа

В данной главе опишем систему терминов и ряд алгоритмов, используемых в данной работе, на которые будет опираться дальнейшее изложение материала. Данный раздел основан на более ранних работах [8, 11, 12, 27, 32–34].

2.1. Основные определения

Прежде всего рассмотрим объекты, которые будут соответствовать переменным, заключающим утверждения. Они образуют *алфавит* — множество, состоящее из атомарных пропозициональных формул (которые могут называться атомами) [27]. $A = \{x_1, ..., x_n\}$ определяет алфавит из n атомов. Для оценки самих атомарных пропозиций, а также связей между ними определим *идеал конъюнктов*, построенный над алфавитом $A = \{x_1, ..., x_n\}$ — множество формул вида $\{x_{i_1}x_{i_2}...x_{i_k} \mid 0 \leq i_1 < ... < i_k \leq n-1, k \leq n\}$, где $x_{i_1}x_{i_2}...x_{i_k}$ представляет конъюнкцию соответствующих переменных [27].

Рассмотренным выше атомам, а также пропозициональным формулам и элементам идеала конъюнктов в частности сопоставим оценки вероятности истинности, которые характеризуют степень уверенности в утверждениях и наборах утверждений, соответствующих атомам и пропозициональным формулам. Таким образом определим объект, расширяющий идеал конъюнктов дополнительными свойствами — *математическую модель фрагмента знаний* (для краткости обозначим как ФЗ) [27]. Математической моделью фрагмента знаний, который построен над алфавитом A, назовем пару (C, p), где C — идеал конъюнктов

над соответствующим алфавитом, p — интервальные или скалярные (точечные) оценки вероятностей для каждого конъюнкта из идеала C [27].

Оценки вероятности истинности для каждого конъюнкта из идеала C, соответствующего некоторому ФЗ, могут быть противоречивыми. Формализуем это свойство. Допустим задан ФЗ (C, p) со скалярными оценками вероятности истинности соответствующих конъюнктов. Тогда этот ФЗ будет *непротиворечив* в том и только том случае, если существует заданная на множестве конъюнктов из ФЗ вероятность p_f, такая что $\forall c \in C, p_f(c) = p(c)$ [33]. Если задан ФЗ с интервальными оценками $(C, \mathbf{p})$, то он будет непротиворечив тогда и только тогда, когда $\forall c \in C$ и $\forall \alpha \in \mathbf{p}(c)$ будет существовать функция $p_{c,\alpha} : C \to [0; 1]$, для которой $p_{c,\alpha}(c) = \alpha$, $\forall x \in C$ выполнено $p_{c,\alpha}(x) \in \mathbf{p}(x))$ и $(C, p_{c,\alpha})$ — непротиворечивый [33].

Для дальнейшей работы с математическими моделями фрагментов знаний и их наборами удобно определить вес, который соответствует каждому ФЗ — *нагрузкой* или *весом* ФЗ $W(C, p)$ назовем подалфавит алфавита, над которым задан ФЗ: $W(C, p) = \{x_i \mid x_i \in C, x_i \in A\}$ [32].

Каждый ФЗ очень тесно характеризует связи между атомарными пропозициями, которые в него входят, и сложность их обработки, как, например, проверка непротиворечивости, растет экспоненциально с увеличением количества атомов. Поэтому, как предполагается в теории алгебраических байесовских сетей, все утверждения разбиваются на группы, фрагменты знаний, имеющие тесные связи между внутренними элементами. Таким образом можно рассматривать наборы фрагментов знаний и, соответственно, наборы моделей этих фрагментов знаний. При этом возможен случай, когда один вес одного ФЗ полностью включается в вес некоторого другого ФЗ и подобные ситуации необходимо исключить ввиду избыточности информации в противном случае. Поэтому назовем *набором максимальных моделей фрагментов знаний* (набор МФЗ, или просто МФЗ, первичная структура алгебраической байесовской сети) такой набор математических моделей фрагментов знаний, что никакая нагрузка ФЗ не содержится полностью в нагрузке другого ФЗ из представленного набора [32]. То есть $\forall i \neq j$ выполнено: $W(V_i) \not\subset W(V_j)$ и $W(V_j) \not\subset W(V_i)$.

Как и отдельные ФЗ, первичная структура также может быть противоречива, причем в таком случае определяется несколько степеней, в частности

- АБС *экстернально непротиворечива*, если каждый ФЗ в АБС непротиворечив и оценки вероятности истинности каждого конъюнкта, входящего в более чем один ФЗ, совпадают с оценками вероятности этого коннъюнкта в других ФЗ [8].
- АБС *интернально непротиворечива*, если каждый ФЗ в АБС непротиворечив, а также для любого конъюнкта из АБС и для любого скалярного значения, взятого из интервала оценки его истинности, есть способ выбрать совпадающие на одинаковых формулах скалярные значения во всех остальных ФЗ в АБС таким образом, что все ФЗ с точечными оценками будут непротиворечивы [8].
- АБС *глобально непротиворечива*, если ее, с ее оценками вероятностей, можно погрузить в объемлющий непротиворечивый ФЗ таким образом, что оценки на формулах АБС не поменяются [8].

После построения алгебраической байесовской сети начнется этап ее применения. На данной стадии возможно изменение изначальных параметров модели, и для этих случаев, в частности, рассматривается такой математический объект, как свидетельство. Он моделирует новую информацию о предметной области. Например, если имеется сформированная модель, описывающая прогноз погоды, то новой информацией будет утверждение о том,

что завтра определенно пойдет снег или то, что завтра, скорее всего, температура будет выше 0 ℃. На основе этой информации необходимо построить свидетельство и учесть его в сформированной модели. Обработка поступающих или оценка возможных для поступления свидетельств является апостериорным выводом. При апостериорном выводе могут рассматриваться три вида свидетельства, детерминированное, стохастическое и неточное [35]:

- *Детерминированным свидетельством* назовем предположение о том, что часть атомов получили конкретное означивание. Обозначается как $\langle c_i, c_j \rangle$, где c_i и c_j — конъюнкты, которые состоят из атомов, получивших соответствующие положительные и отрицательные означивания [35].
- *Стохастическое свидетельство* — предположение о том, что над C', подыдеале C, задается непротиворечивый ФЗ со скалярными оценками, определяющий вероятности истинности соответствующих конъюнктов [35].
- *Неточное свидетельство* — предположение о том, что над C', подыдеале C, задается непротиворечивый ФЗ с интервальными оценками, который определяет вероятности истинности соответствующих элементов подыдеала [35].

Данные свидетельстваа можно локально распространять (пропагировать) в моделях фрагментов знаний, что описано в [35]. Также пропагацию свидетельства можно проводить и глобально, на всей алгебраической байесовской сети [8].

2.2. Вторичная структура АБС

Помимо рассмотрения простых наборов ФЗ существует способ их представления в виде графовой структуры, дополняющей модели фрагментов знаний связями между ними. Для изучения такого способа сперва дадим понятие *сепаратора*, общего для двух ФЗ. То есть сепаратором двух МФЗ, V_i и V_j, назовем *подалфавит*, который является пересечением нагрузок этих ФЗ: $W(V_i, V_j) = W(V_i) \cap W(V_j), i \neq j$ [12]. При этом пара МФЗ называются *сочлененными*, если их сепаратор непуст [12].

Новой глобальной структурой будет являться в таком случае *граф максимальных моделей фрагментов знаний* — ненаправленный граф, вершинам которого сопоставлены МФЗ, вошедшие в АБС и ребра возможны только между сочлененными ФЗ. Как и для модели фрагмента знаний, для ребра этого графа также удобно дать понятие веса [32]. *Нагрузкой* $W(\{V_i, V_j\})$ *ребра* $\{V_i, V_j\} \in E(G)$ графа G назовем сепаратор его концов: $W(\{V_i, V_j\}) = W(V_i) \cap W(V_j)$ [12]. Определим и *нагрузку* $W(H)$ *подграфа* $H \subseteq G$ — наибольший по включения подалфавит, входящий в нагрузку всех вершин подграфа: $W(H) = \bigcap_{V \in H} W(V)$ [12].

Для графов, построенных по вышеописанному принципу, характерен определенного вида путь. А именно рассмотрим *магистральный путь* между сочлененными вершинами V_i и V_j — такой путь между этими вершинами, что нагрузка каждой вершины пути содержит сепаратор концов этого пути [32]. Далее граф будет *магистрально связен*, если между каждой из его сочлененных вершин существует магистральный путь [32]. *Граф смежности* — магистрально связный граф МФЗ [32]. *Дерево смежности* — граф смежности, представимый в виде дерева [32].

Если имеется ацикличная АБС, то из ее интернальной непротиворечивости следует глобальная [8]. Заметим, что с учетом этого утверждения экстернально непротиворечивая АБС со скалярными оценками, представимая в виде дерева смежности (ацикличная АБС), глобально непротиворечива, так как экстернальная непротиворечивость для такой АБС влечет интернальную непосредственно по определению. Для поддержания и проверки интерналь-

ной, глобальной непротиворечивости АБС, непротиворечивости ФЗ применяется решение задач линейного программирования (ЗЛП) [8, 27], при этом проверка интернальной непротиворечивости (а тем более экстернальной) имеет значительно меньшую вычислительную сложность, чем проверка глобальной, поэтому утверждение о глобальной непротиворечивости, следуемой из интернальной (экстернальной) полезно при практической реализации проверки непротиворечивости [8].

В результате, помимо первичной структуры АБС, можно дать определение *вторичной*. Такой структурой будет являться некоторый граф смежности АБС [8].

Рассмотрим частные случаи вторичных структур. Например, *минимальный граф смежности* (МГС) — граф смежности, число ребер которого минимально [32].

Так же существует понятие *максимального графа смежности* G_{max} — наибольшего по числу ребер графа смежности [32]. Для заданного множества вершин существует единственный максимальный граф смежности, то есть тот, в котором между вершинами существует ребро только тогда, когда они сочлененные [32].

Следуя [34], дополнительно предположим, что первичная структура *связна*, то есть связен максимальный граф смежности, построенный над этой структурой. В противном случае можно рассматривать наборы вершин из каждой компоненты связности как отдельные АБС.

2.3. Третичная структура АБС

Далее перейдем к основному объекту, рассматриваемому в работе. Перед его непосредственным определением необходимо также введение нескольких новых понятий. Так, *сужением* $G \downarrow U$ графа G на нагрузку U назовем граф, в который входят те и только те ребра и вершины исходного графа G, нагрузки которых равны или содержат U [32]. *Значимое сужение* — сужение на нагрузку, которая является сепаратором для некоторой пары МФЗ [32]. На сужение можно наложить дополнительные ограничения, тогда получим *сильное сужение* $G \Downarrow U$ — сужение $G \downarrow U$, из которого удалили все ребра нагрузки U [34]. После сильного сужения граф $G \downarrow U$ разбивается на компоненты связности, после сужения же $G \downarrow U$ граф остается связным [34].

Одним из основных объектов в новоопределяемой структуре будет *значимая нагрузка* U — непустой сепаратор некоторой пары ФЗ первичной структуры [12]. *Замкнутым же снизу множеством нагрузок* назовем объединение множества значимых нагрузок с множеством нагрузок вершин МФЗ [12]. *Замкнутое множество нагрузок* — объединение замкнутого снизу множество нагрузок с одноэлементным множеством, содержащим пустое множество [12].

При этом на множестве нагрузок существует частичный порядок, являющийся отношением включения. Таким образом, *родительским графом* (*третичной структурой* АБС) назовем диаграмму Хассе замкнутого множества нагрузок [12]. Диаграмму Хассе можно рассматривать как транзитивное сокращение, поэтому родительский граф единственный при заданной первичной структуре АБС [12, 36].

3. Апостериорный вывод

В этом разделе опишем алгоритм использования третичной структуры в апостериорном выводе. В итоге получим основную процедуру `PosterioriInfer`, при применении которой будет пропагироваться соответствующее свидетельство.

Общее описание алгоритма апостериорного вывода с применением третичной структуры рассматривается в работе [12]. В ней доказывается теорема о том, что алгоритм завершит работу и после завершения строит оценки вероятностей, которые в случае точечных оценок совпадают с результатом работы алгоритма пропагации свидетельства по минимальному графу смежности, при этом рассматриваются только ациклические алгебраические сети. Рассмотрим это общее описание, взятое из [12]:

1. Распространить свидетельство в ФЗ, входящие в сужение $G_{max} \downarrow u$ для некоторой нагрузки u, содержащей нагрузку свидетельства. Повторная пропагация в ФЗ не проводится [12].
2. Пометить u и всех ее потомков. Если существуют нагрузки со всеми помеченными детьми, пометить их [12].
3. Выбрать непомеченную нагрузку максимальной мощности v (по количеству атомарных пропозиций), сыном которой является некоторая помеченная нагрузка w. Если все нагрузки помечены и выбрать v не удается, то завершить работу [12].
4. В противном случае, если удалось выбрать v, сформировать свидетельство нагрузка которого совпадает с v, а оценки вероятности взяты из какого-либо ФЗ w. Перейти к шагу 1 [12].

В работе также описывается, что данный алгоритм выполним, если нагрузка изначального свидетельства содержится хотя бы в одном узле родительского графа, иначе применяют метод распространения множества детерминированных свидетельств, рассматриваемый в работе [8].

Этот алгоритм (псевдокод), а также вспомогательные алгоритмы, представлены в листингах 1 и 2.

Пояснения к алгоритмам:

- Алгоритм `InferInSubgraph`
  - `Mark` и `IsMarked` — помечающие нагрузку и проверяющие на наличие метки процедуры;
  - `Propagate` — процедура, локально распространяющая свидетельство.
- Алгоритм `GenerateEvidenceIfPossible`
  - `SeparatorsNarrowing` — словарь, сопоставляющий каждому сепаратору ФЗ, входящих в сужение на него;
  - `GenerateEvidence(u, kp)` — функция, формирующая свидетельство для конъюнктов из `u` и оценок из `kp`;
  - `Any(s)` — функция, возвращающая некоторый (любой) элемент из множества s;
  - Можно увеличить скорость работы алгоритма, если при установлении метки оповещать родительские вершины об этом, а при просмотре родительских вершин просматривать количество меток у детей. Но для этого могут понадобиться дополнительные ссылки на родителей.
- Алгоритм `PosterioriInfer`
  - `SortedWeights` — все нагрузки, включая ФЗ и сепараторы, отсортированные в порядке уменьшения количества атомарных пропозиций. В конце стоит пустая нагрузка — вершина родительского графа;
  - `Contains` — функция, проверяющая включение свидетельства во ФЗ. Работает за $O(1)$.

Листинг 1 Вспомогательные алгоритмы, пропагирующие свидетельство в подграфе (`InferInSubgraph`) и формирующие свидетельство (`GenerateEvidenceIfPossible`) в соответствии с алгоритмом апостериорного вывода

```
1: ▷ Реализует шаг 1 и часть шага 2
2: procedure InferInSubgraph(e ∈ Evidences, w ∈ Weights)
3:     Mark(w)                                  ▷ В процессе пропагации помечаем вершины
4:     if w ∈ KnowledgePatterns then
5:         Propagate(e, w)                      ▷ Распространяем свидетельство в ФЗ
6:         return                               ▷ Вершина с нагрузкой в виде ФЗ — лист, поэтому потомков у нее нет. Эта строка для ясности алгоритма, без нее следующий цикл все равно бы не запустился
7:     for all child ∈ w.Children do            ▷ Распространяем свидетельство вниз по родительскому графу — от родителей к потомкам
8:         if not IsMarked(child) then          ▷ Повторная пропагация не проводится
9:             InferInSubgraph(e, child)
10:
11: ▷ Формирует свидетельство в 4 шаге. Свидетельсто формируется только тогда, когда у непомеченой v есть и помеченный, и непомеченный потомок. В противном случае v не подходит и возвращается Null
12: function GenerateEvidenceIfPossible(v ∈ Weight, sn ∈ SeparatorsNarrowing)
13:     if not IsMarked(v) then                  ▷ Сразу проверяем v на непомеченность
14:         hasMarkedChild ← false
15:         hasNotMarkedChild ← false
16:         kpWithEvidence ← Null                ▷ kpWithEvidence будет хранить ФЗ для w из 4-го шага
17:         for all w ∈ v.Children do
18:             if IsMarked(w) then
19:                 if not hasMarkedChild then
20:                     hasMarkedChild ← true
21:                     kpWithEvidence ← Any(sn[w])   ▷ Выбираем некоторый (любой) ФЗ, входящий в сужение на w
22:             else
23:                 hasNotMarkedChild ← true
24:             if hasMarkedChild and hasNotMarkedChild then
25:                 return GenerateEvidence(v, kpWithEvidence)
26:         if hasMarkedChild then
27:             Mark(v)                          ▷ Будут помечены все сыновья, так как hasNotMarkedChild = false, иначе возвратилось бы свидетельство в строке 24
28:     return Null
```

Листинг 2 Алгоритм апостериорного вывода с применением третичной структуры `PosterioriInfer`

```
1: ▷ Процедура, при применении которой пропагируется свидетельство e
2: procedure PosterioriInfer(e ∈ Evidences, pg ∈ ParentGraphs)
3:     weightToPropagate ← Null                          ▷ weightToPropagate будет хранить нагрузку для следующей пропагации
4:     for all w ∈ Reversed(pg.SortedWeights) do         ▷ Ищем начальную нагрузку для пропагации — наименьшую по включению нагрузку, содержащую e
5:         if Contains(w, e) then
6:             weightToPropagate ← w
7:             break
8:     if weightToPropagate = Null then                  ▷ В алгоритме рассматривается только случай, когда свидетельство содержится хотя бы в одной нагрузке
9:         return
10:    while not IsMarked(pg.EmptyWeight) do             ▷ Пока не помечен весь граф (если помечена пустая нагрузка, то помечены все ее потомки, а значит и весь граф)
11:        InferInSubgraph(e, weightToPropagate)         ▷ Распространяем свидетельство в подграф с корнем в выбранной нагрузке
12:        for all v ∈ pg.SortedWeights do               ▷ Первая часть 3-го шага. Выбираем нагрузку v
13:            e ← GenerateEvidenceIfPossible(v, pg.SortedWeights)   ▷ Формируем новое свидетельство (в 4 шаге)
14:            if e ≠ Null then
15:                weightToPropagate ← v                 ▷ Выбираем v
16:                break
17:    if pg.HasIntervalKP or e.IsInterval then
18:        KeepInternalConsistency(pg)                   ▷ Если оценки получились интервальными, то поддерживаем интернальную непротиворечивость
```

Утверждение 1. Алгоритм `PosterioriInfer` в случае скалярных оценок назначает оценки вероятностей, совпадающие с результатом пропагации по МГС, а в случае интервальных оценок строит накрывающие оценки, которые бы соответствовали результату пропагации в МГС, если бы распространение виртуального свидетельства давало точные оценки.

Доказательство. Для начала рассмотрим соответствие представленного алгоритма и общего описания. В целом, происходит непосредственная реализация общего описания, с тем лишь отличием, что установление метки на вершины, для которых все сыновья помечены, происходит во время выбора вершины u в шаге 3. Алгоритм отработает корректно, так как сыновья включают большее число атомарных пропозиций, поэтому они, рассматривая `sortedWeights`, будут правильно помечены перед рассмотрением родительской вершины.

Случай скалярных оценок разобран в работе [12], показывающий, что при данных условиях алгоритм строит оценки вероятностей, которые совпадают с результатом пропагации свидетельства по минимальному графу смежности.

Разберем работу алгоритма с интервальными оценками вероятности истинности. В таком случае пропагация виртуального свидетельства между двумя ФЗ предполагает нахождение экстремальных значений, сопоставляемых итоговым оценкам, получаемым после пропагации. При этом при нахождении соответствующих минимумов и максимумов рассматриваются все элементы из семейств распределений, соответствующих оценкам как свидетельства, так и самого ФЗ [37]. Поэтому, если свидетельство и/или ФЗ имели накрывающие интервальные оценки, то оценки вероятности истинности, полученные в таком случае после распространения свидетельства, будут накрывающими по отношению к оценкам, полученным в результате пропагации с точными оценками.

Далее, как показывается в работе [12], после шага 2 подграф G' минимального графа смежности G, состоящий из помеченных ФЗ, остается связен, а подграф, получаемый из непомеченных вершин, разбивается на несколько компонент связности, и пропагация в МГС в каждую компоненту связности c_i в происходит по единственному ребру, соединяющему G' и c_i. В таком случае пропагация свидетельства на новые компоненты связности c_i в МГС будет соответствовать пропагации с использованием алгоритма с тем лишь отличием, что могут использоваться разные накрывающие оценки для виртуальных свидетельств и получаться таким образом различные накрывающие апостериорные оценки.

Так же стоит добавить, что в результате пропагации по алгоритму могут получится интернально противоречивые накрывающие оценки, поэтому необходимо обеспечивать соответствующую непротиворечивость, которая, в случае ацикличной первичной структуры даст глобальную. □

Утверждение 2. Сложность работы алгоритма апостериорного вывода с применением третичной структуры в случае скалярных оценок лежит в классе $O(w(w \cdot O(\texttt{Propagate}) + s_{all}e_s + c))$, где w — общее количество ФЗ, $O(\texttt{Propagate})$ — сложность функции, локально распространяющей свидетельство, s_{all} — количество вершин в родительском графе, e_s — количество ребер в графе смежности, c — максимальное число конъюнктов в ФЗ.

Доказательство. Рассмотрим сложность `InferInSubgraph`. Алгоритм в строке 7 рассматривает все ребра в подграфе родительского графа, содержащие нагрузку u, в количестве e_u. Дойдя до листьев-ФЗ, рассматриваемых в количестве w_u штук, алгоритм пропагирует в них свидетельство (строка 5). Таким образом сложность алгоритма будет выражаться как $O(w_u \cdot O(\texttt{Propagate}) + e_u)$. При этом $w_u \leq w$ и $e_u \leq e_s$, отсюда следует более общий класс, в котором лежит сложность `InferInSubgraph`, равный $O(w \cdot O(\texttt{Propagate}) + e_s)$.

Далее оценим сложность алгоритма `GenerateEvidenceIfPossible`. Алгоритм рассматривает всех детей вершины u в количестве ch_u (строка 17) и, возможно, генерирует свидетельство (строка 25), беря оценки из ФЗ KPWithEvidence. Взятие оценок потребует не больше чем c операций, и $ch_u \leq e_s$. Поэтому сложность алгоритма лежит в классе $O(e_s + c)$.

Перейдем к алгоритму `PosterioriInfer`. В строке 4 рассматриваются не более чем s_{all} элементов, где s_{all} — количество вершин в родительском графе. Далее заметим, что на каждом шаге цикла while происходит пропагация свидетельства хотя бы в один новый ФЗ, поэтому количество итераций этого цикла не будет превосходить w. На каждом шаге происходит работа алгоритма `InferInSubgraph`, затем выполняется не более чем s_{all} раз работа `GenerateEvidenceIfPossible`. Замечу, что создание нового свитедельства в `GenerateEvidenceIfPossible` происходит один раз за итерацию цикла while, поэтому оцен-

ку сложности всего алгоритма можно представить в виде $O(s_{all} + w(w \cdot O(\texttt{Propagate}) + e_s + s_{all}e_s + c))$, что лежит в классе $O(w(w \cdot O(\texttt{Propagate}) + s_{all}e_s + c))$. □

В случае интервальных оценок добавляется сложность поддержания интернальной непротиворечивости.

4. Проверка ацикличности

В этом разделе решим задачу проверки ацикличности алгебраической байесовской сети с использованием только третичной структуры. В результате опишем функцию `GetComponentQuantity`, после применения которой получим общее число компонент связности, что в сочетании со следующей теоремой, описанной в [31], поможет дать ответ на вопрос об ацикличности алгебраической байесовской сети:

Теорема 1. Связная первичная структура АБС циклична тогда и только тогда, когда не выполняется соотношение:

$$|\text{МКР}| = \sum_{U \in Sep} Conn(G_{max} \downarrow U) - |Sep| + 1,$$

где МКР — первичная структура АБС, набор ФЗ, $Conn(G_{max} \downarrow U)$ — число компонент связности графа $G_{max} \downarrow U$, Sep — множество непустых сепараторов.

Для того, чтобы применить эту теорему на практике, необходимо рассчитать каждое из слагаемых выражения. Мощность первичной структуры равна количеству листьев-ФЗ в родительском графе, что можно посчитать при построении родительского графа. Количество вершин родительского графа будет соответствовать количеству сепараторов, исключая пустую, верхнюю вершину и листья. Таким образом, остается получить число компонент связности с использованием третичной структуры. Для этого, сперва, докажем теорему:

Теорема 2. Возьмем пару вершин с нагрузками в виде ФЗ kp_1 и kp_2 и сильное сужение на значимую нагрузку u. Тогда эти вершины лежат в одной компоненте связности C_u графа $G_{max} \downarrow u$ в том и только том случае, если существует такая последовательность вершин с нагрузками в виде ФЗ, что вершины с нагрузками kp_1 и kp_2 являются крайними, а также для любой пары соседних вершин v_i и v_{i+1} в этой последовательности существует нагрузка, которая является одновременно и предком по отношению к v_i и v_{i+1}, и потомком по отношению к вершине с нагрузкой u в родительском графе. Формально:

$$kp_1, kp_2 \in C_u \Leftrightarrow \exists v_1 = kp_1, v_2, ..., v_{n-1}, v_n = kp_2 :$$

$$\forall i = 1, ..., n-1 : (\exists w : v_i, v_{i+1} \in \text{descendants}(w) \;\&\; w \in \text{descendants}(u)).$$

Доказательство. Пусть нашлась последовательность как в условии теоремы, связывающая kp_1 и kp_2 в графе $G_{max} \downarrow u$. Рассмотрим пару соседних элементов последовательности. Так как между этими элементами существует вершина w, которая является предком по отношению к ним, то ребро, соединяющее эти элементы, включает вес w. С другой стороны, w — потомок по отношению к u, поэтому нагрузка связующего ребра будет содержать, но не равняться u. Таким образом, существует путь в $G_{max} \downarrow u$ между вершинами с нагрузками kp_1 и kp_2, поэтому они лежат в одной компоненте связности C_u.

Пусть kp_1 и kp_2 лежат в одной компоненте связности C_u. Тогда между ними существует путь, который связан ребрами, нагрузки которых содержат, но не равняются нагрузке сильного сужения u. Следовательно, существует указанная в условии теоремы последовательность вершин. □

Замечание 1. Рассмотрим две вершины w_1 и w_2 в родительском графе. Если существует вершина v, являющаяся для них общим потомком, то все ФЗ, которые являются потомками по отношению к w_1 и w_2, лежат в одной компоненте связности, которая была получена в результате сильного сужения на новую, общую для вершин w_1 и w_2 вершину-предка u. Более формально: если $\exists w_1, w_2 : kp_1 \in \text{descendants}(w_1)$ & $kp_2 \in \text{descendants}(w_2)$ & $w_1, w_2 \in \text{descendants}(u)$, а также $\exists v : v \in \text{descendants}(w_1) \cap \text{descendants}(w_2)$, то $kp_1, kp_2 \in C_u$.

Таким образом, предоставим алгоритм подсчета числа компонент связности, опирающийся на доказанную теорему. Предположим, что нужно найти число компонент связности графа $G_{max} \downarrow u$. Для этого будем распространять по родительскому графу от вершины u маркеры, назовем их *цветами*. Сначала распространим от u *различные* цвета до каждой вершины-потомка u. Далее, от каждой вершины-потомка u распространим по уже их вершинам-потомкам цвет, который был получен ранее. В том случае, если в некоторую вершину поступало несколько отличающихся цветов, то тогда примем эти цвета одинаковыми. В конечном счете рассмотрим количество различных цветов, которые остались после распространения цветов до листьев родительского графа. Это количество и будет соответствовать количеству компонент связности $G_{max} \downarrow u$. Пример распространения цветов показан на рис. 1.

Покажем, что алгоритм корректен. Предположим, что нашлись две вершины с нагрузками в виде ФЗ, которые получили одинаковый цвет. Тогда, исходя из замечания 1, эти вершины будут лежать в одной компоненте связности $G_{max} \downarrow u$. Далее допустим, что некоторые две вершины родительского графа с нагрузками kp_1 и kp_2 получили различные цвета, но тем не менее лежат в единой компоненте связности. В этом случае, в соответствии с теоремой 2, должна существовать последовательность из вершин $v_1, ..., v_n$, где каждая пара соседних вершин имеет по построению одинаковый цвет. Но тогда $v_1 = kp_1$ и $v_n = kp_n$ будут окрашены в один цвет — противоречие.

При реализации рассмотренного алгоритма возникает вопрос о том, как эффективно хранить и объединять цвета. В работе предлагается распространять вместо цветов числа вида $2^i, i = 0, ..., n-1$, где n — число дочерних узлов начальной вершины, рассматриваемой для распространения. В таком случае объединение цвета — применение побитового «или».

Далее необходимо понять то, как будут распространяться цвета. Предположим, что мы будем распространять от каждой вершины все цвета, которые могли дойти до этой вершины. В противном случае было бы необходимо для каждой вершины хранить отдельно множества цветов и каким-то образом объединять схожие цвета, перебирать эти множества на каждом шаге распространения. Такой алгоритм интуитивно более ресурсозатратен и его изучение не вошло в рамки данной статьи.

Итак, если использовать для распространения цвета, например, обход в ширину, то может произойти ситуация, при которой передача цветов произойдет до того, как вершина получит информацию о всех приходящих цветах, как в примере на рис. 2. В данной работе предлагается при построении родительского графа хранить в вершинах число родителей и

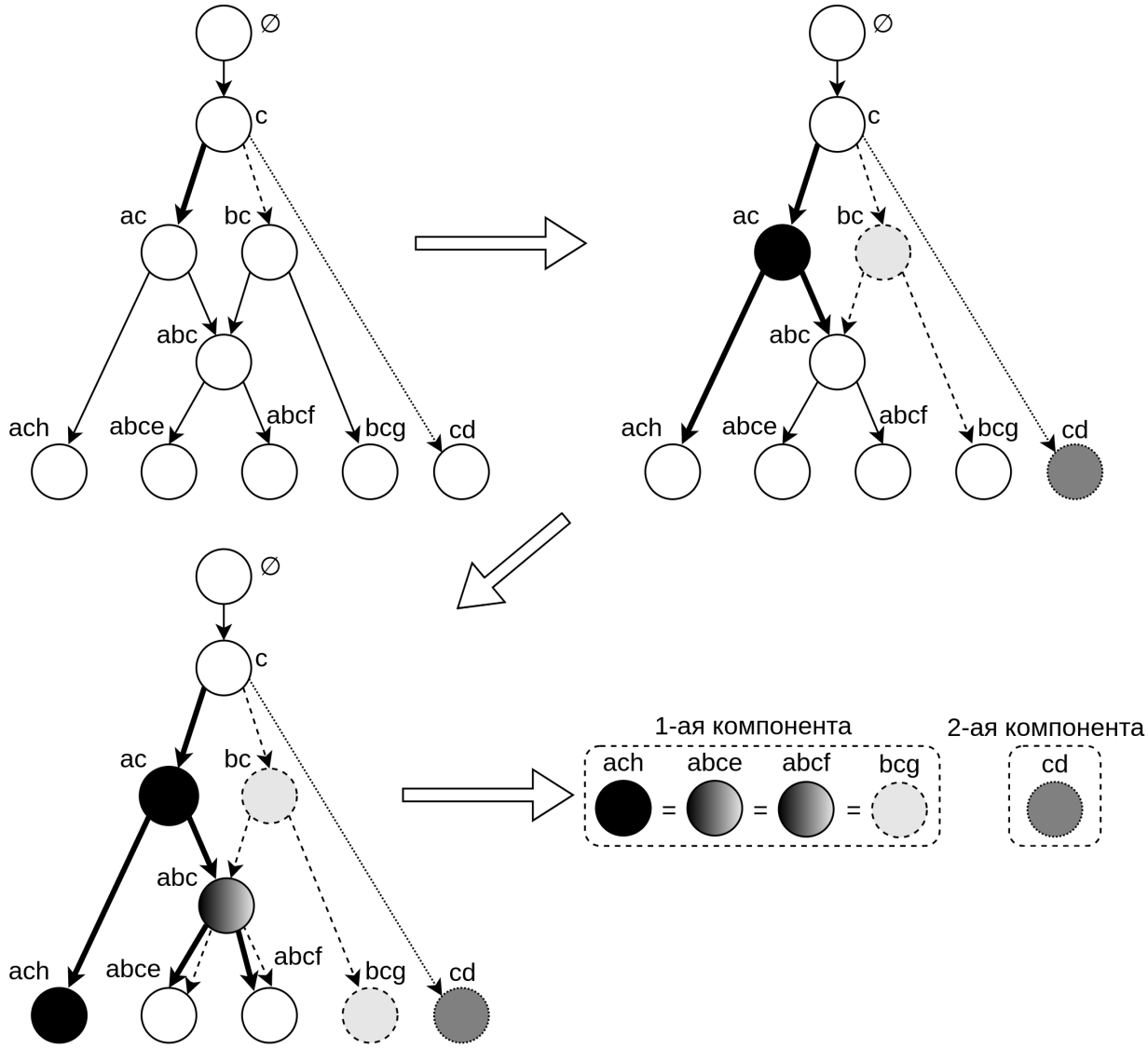


Рис. 1. Распространение цветов при сужении на вершину c. Всего три цвета, вершины помечаются различными цветами, ребра — толщиной ребра и пунктиром с отличающимся интервалом между штрихами

проводить распространение цвета только после того, как вершину просмотрели все родительские узлы. Алгоритм распространения цветов `SpreadColors` представлен в листинге 3.

Пояснения к `SpreadColors`:

- `WeightsToColors` — словарь, сопоставляющий цвет сепаратору. Относительно каждого сепаратора-предка вершина будет иметь свой цвет, который будет передаваться дочерним вершинам. Его можно удалять у вершины, если полностью произошло распространение цветов к дочерним узлам;
- `NumberOfParents` задается при построении родительского графа, `ParentCount` так же можно инициализировать нулем при построении графа;
- & и | — побитовые операции «и» и «или» соответственно.

Утверждение 3. В результате работы процедуры `SpreadColors` каждый ФЗ получит корректное распределение цветов.

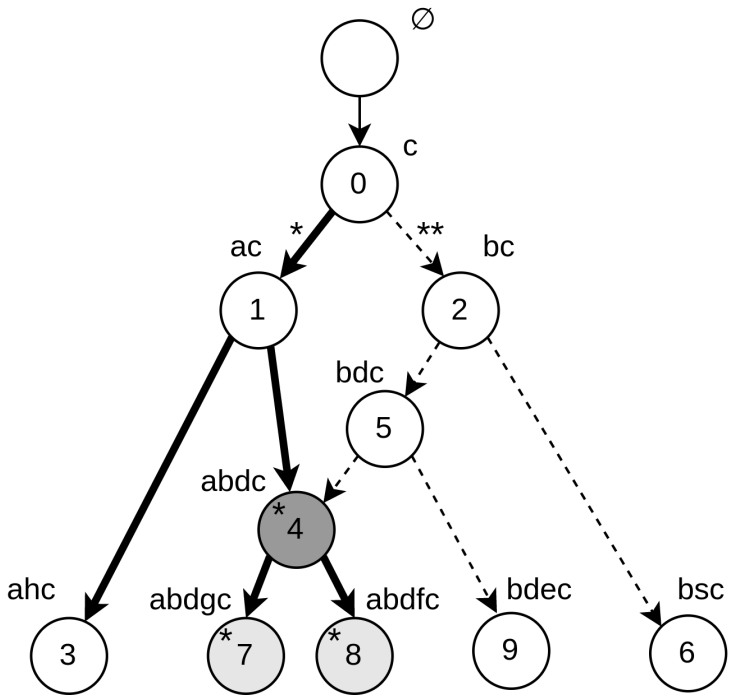


Рис. 2. Возникновение ошибки распространения цвета (4-я вершина) при обходе в ширину. Всего два цвета, они помечены как «*» и «**». Вершина №4 должна распространить оба цвета, но продвигается только «*». Номера вершин соответствуют порядку обхода

Доказательство. Действительно, реализуется непосредственно вышеописанный алгоритм, при этом цвет распространяется от вершины только тогда, когда получены цвета от всех родительских вершин. □

Утверждение 4. Сложность работы процедуры `SpreadColors` лежит в классе $O(e_s s)$, где e_s — количество ребер в родительском графе, а s — количество непустых сепараторов.

Доказательство. В строке 4 проверяются все дочерние узлы текущей вершины, при этом каждая вершина рассматривается как родительская только один раз. Соответственно, таких проверок будет e_s штук. При этом в 5 строке при каждой проверке перебираются все все сепараторы из `WeightsToColors`, которых может быть не более s. Все остальные шаги алгоритма имеют сложность класса $O(1)$, из чего следует искомая сложность алгоритма. □

Конечным этапом будет последнее объединение цветов, дошедших до листьев-ФЗ. Его идея заключается в том, что сначала формируется список наборов цветов, дошедших до ФЗ и соответствующих каждому сепаратору. Затем каждый представляемый в виде числа набор цветов, продвигаясь по списку, сравнивается с остальными наборами и объединяется в том случае, если есть хотя бы одно пересечение, при этом использованные наборы цветов маркируются. Далее последний объединяемый набор помечается особым образом и на его место в списке записывается совокупность цветов, полученных при текущей итерации объединений. Записываемые после распространения цвета, *последние цвета*, могут далее участвовать в объединении, в отличие от промаркированных простым образом. В итоге, после работы такого алгоритма количество последних цветов и будет соответствовать ко-

Листинг 3 Алгоритм распространения цветов до листьев-ФЗ `SpreadColors`

```
1: procedure SpreadColors(w ∈ Weights)
2:     if w ∈ KnowledgePatterns then
3:         return                                  ▷ Вершина с нагрузкой в виде ФЗ — лист, поэтому
                                                     потомков у нее нет, распространять цвета даль-
                                                     ше не нужно. Эта строка для ясности алгорит-
                                                     ма, без нее следующий цикл все равно бы не
                                                     запустился
4:     for all v_i ∈ w.Children do                 ▷ Перебираем дочерние вершины для распростра-
                                                     нения цветов
5:         for all s ∈ w.WeightsToColors do        ▷ Рассматриваем все сепараторы, относительно
                                                     которых нагрузка w получила цвета
6:             if s ∈ v_i.WeightsToColors then     ▷ Если v_i ранее получала цвет относительно сепа-
                                                     ратора s
7:                 ▷ То добавляем цвет от w
8:                 v_i.WeightsToColors[s] ← v_i.WeightsToColors[s] | w.WeightsToColors[s]
9:             else
10:                ▷ В противном случае кладем к v_i новый цвет
11:                v_i.WeightsToColors[s] ← w.WeightsToColors[s]
12:        v_i.WeightsToColors[w] ← 2^i            ▷ В каждую v_i кладем свой цвет относительно w
13:        v_i.ParentCount ← v_i.ParentCount + 1   ▷ Учитываем, что для одного из родителей v_i про-
                                                     изошла передача цвета
14:        if v_i.NumberOfParents = v_i.ParentCount then
15:            SpreadColors(v_i)                   ▷ Продолжаем распространять цвета от дочерних
                                                     вершин только в том случае, когда были полу-
                                                     чены цвета от всех родителей
```

личеству компонент связности. Этот алгоритм (`GetComponentQuantity`) описан в листинге 4, пример его работы изображен на рис. 3.

Пояснения к `GetComponentQuantity`:

- `GetListOfColors` — функция, получающая список итоговых наборов цветов, дошедших до ФЗ;
- `SimplyMark(i)` и `MarkAsLast(i)` — взаимозаменяющие процедуры, помечающие элемент с индексом i самого списка, `IsSimplyMarked` и `IsMarkedAsLast` — процедуры, проверяющие то, как помечены элементы. Могут быть реализованы через создание массива, по количеству элементов равному количеству ФЗ, которые рассматриваются для текущего сепаратора, и хранящего соответствующие маркировки.
- `QuantityOfMarkedAsLast` — функция, возвращающая количество помеченных как последние элементов. Может реализовываться, например, через перебор массива.

Утверждение 5. В результате работы функции `GetComponentQuantity` возвратится общее количество компонент связности.

Доказательство. Начнем с того, что процедура `SpreadColors` корректно распределит цвета по листьям-ФЗ. Далее заметим, что каждый из начальных наборов цветов списка (где набор цветов — сумма соответствующих чисел 2^i, дошедших до ФЗ) будет полностью лежать

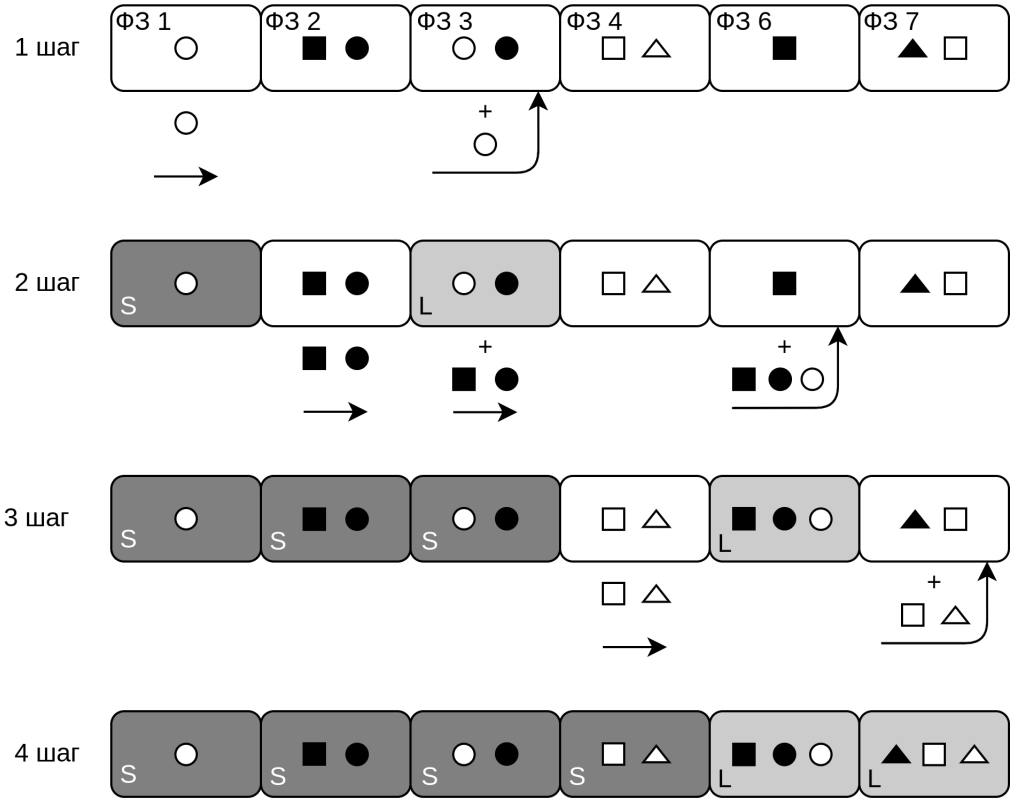


Рис. 3. Часть работы алгоритма `GetComponentQuantity` с вычислением последних объединений цветов для одного из сепараторов, где в итоге получается 2 компоненты связности. Цвета из алгоритма обозначены фигурами различной формы и цвета. Темно-серый цвет с пометкой «S» — «простая» маркировка, светло-серый цвет с пометкой «L» — «последние» элементы

в некотором элементе, помеченном как последний. Это основывается на том, что каждая вершина, соответствующая `IsSimplyMarked`, могла быть помечена только при объединении с каким-либо распространяющимся набором цветов, положенным в итоге в некоторую последнюю вершину.

Назовем путем наборов цветов такую последовательность изначальных наборов цветов, что каждая пара соседних элементов имеет одинаковый цвет. Заметим также, что если два первоначальных цвета лежат в последнем элементе, то, по построению, существует путь наборов цветов, который соединяет эти два цвета (то есть первый цвет лежит в первом наборе, а последний — в последнем).

С другой стороны, предположим, что существует путь, соединяющий некоторые два цвета. Каждая пара соседних наборов в этом пути должна лежать в одном и только одном последнем элементе, ведь при распространении общего для них цвета объединяющий набор цветов ($color_i$ в алгоритме) соединит эти наборы (даже если один уже был в некоторой последнем элементе) и добавит их в новый последний элемент. Объединение произойдет, так как только один из них не может быть промаркирован простым способом. При этом, каждый набор цветов попадет только в один последний элемент, так как он после объединения маркируется и не учитывается более. Итак, каждая пара наборов цветов будет лежать в одном последнем элементе, поэтому и весь путь будет лежать в нем.

Листинг 4 Алгоритм вычисления количества компонент связности для всех сильных сужений на сепараторы `GetComponentQuantity`

```
1: function GetComponentQuantity(pg ∈ ParentGraphs)
2:     SpreadColors(pg.MainNode)                          ▷ Распространяем цвета перед тем, как посчитать различные цвета в листьях-ФЗ
3:     totalComponentQuantity ← 0                         ▷ Будет хранить общее число компонент связности
4:     for all s ∈ Separators do                          ▷ Перебираем все сепараторы для определения количества различных цветов
5:         colors ← GetListOfColors(s, pg.KnowledgePatterns)   ▷ Получаем цвета, дошедшие до листьев-ФЗ
6:         for all color_i ∈ colors do                    ▷ Будем рассматривать цвета, идя вперед по массиву цветов и объединяя одинаковые
7:             if not IsSimplyMarked(i) and
                  not IsMarkedAsLast(i) then              ▷ Рассматриваем только непромаркированные элементы, так как для маркированных элементов распространение цвета вперед по массиву цветов уже произошло
8:                 SimplyMark(i)                          ▷ Помечаем как рассмотренный (*простым* образом)
9:                 lastHandled ← i                        ▷ Хранит индекс последнего элемента, с которым произошло объединение
10:                for all color_j ∈ colors[j > i] do
11:                    if color_i & color_j ≠ 0 and
                          not IsSimplyMarked(j) then      ▷ Если цвета одинаковые и если элемент не промаркирован *простым* образом (если элемент промаркирован простым образом, то нет смысла его рассматривать, так как цвет, который он содержит, будет содержаться в некотором *последнем* элементе, идущем далее), то
12:                        color_i ← color_i | color_j    ▷ Объединяем цвета. Меняется только переменная color_i, но не элемент массива
13:                        SimplyMark(j)
14:                        lastHandled ← j
15:                MarkAsLast(lastHandled)                ▷ Отдельно помечаем *последний* элемент. Их количество будет соответствовать количеству компонент связности
16:                colors[lastHandled] ← color_i          ▷ Записываем объединенный цвет, далее он также может сравниваться и объединяться
17:        totalComponentQuantity ← totalComponentQuantity +
               QuantityOfMarkedAsLast(colors)             ▷ Увеличиваем счетчик общего числа компонент связности, рассчитав количество компонент для рассмотренных на данном шаге цветов
18:    return totalComponentQuantity
```

Суммируя, получаем, что если два цвета должны быть связаны, то они не будут лежать в разных последних элементах, а будут лежать в одном, и при этом каждые два цвета в последнем элементе будут связаны, из чего следует доказательство утверждения. □

Утверждение 6. Сложность работы функции `GetComponentQuantity` лежит в классе $O(sw^2 + e_s s)$, где w — общее количество ФЗ.

Доказательство. Сначала заметим, что сложность работы `SpreadColors` лежит в классе $O(e_s s)$, по утверждению 4. Далее, в строке 4 перебираются все сепараторы, количество которых — s. Для каждого сепаратора в строке 6 рассматриваются все ФЗ, до которых дошел цвет этого сепаратора. Затем для этого цвета вновь просматриваются наборы цветов, которых не более чем w. Сложность работы функций `GetListOfColors` и `QuantityOfMarkedAsLast` лежит, как максимум, в классе $O(w)$. Все остальные шаги имеют сложность $O(1)$. В итоге, суммируя, получаем искомую сложность. □

Стоит отметить, что при небольших модификациях и дополнениях к данному алгоритму можно получить информацию о сужениях и сильных сужениях на все значимые нагрузки, которая представлена в соответствующих наборах цветов, принадлежащих ФЗ.

5. Результаты и обсуждение

В данной работе были представлены алгоритмы, позволяющие применять только третичную структуру АБС в глобальном апостериорном выводе. Так непосредственно был расширен до возможности применения в случае интервальных оценок существующий до написания работы алгоритм [12], способный ранее производить глобальный апостериорный вывод только в случае скалярных оценок. Это было достигнуто за счет поддержания интернальной непротиворечивости. Также алгоритм описан более подробно, то есть представлен в виде псевдокода. Таким образом, создана процедура `PosterioriInfer`, при применении которой будет пропагироваться соответствующее свидетельство. Процедура `PosterioriInfer` использует вспомогательную функцию `GenerateEvidenceIfPossible`, формирующее свидетельство, а также процедуру `InferInSubgraph`, распространяющую свидетельство в подграф родительского графа. Доказана корректность работы расширенного алгоритма, а именно показано, что процедура `PosterioriInfer` в случае скалярных оценок назначает оценки вероятностей, совпадающие с результатом пропагации по МГС, а в случае интервальных оценок строит накрывающие оценки, которые бы соответствовали результату пропагации в МГС, если бы распространение виртуального свидетельства давало точные оценки. Доказана сложность работы процедуры `PosterioriInfer`, которая в случае применения скалярных оценок лежит в классе $O(w(w \cdot O(\texttt{Propagate}) + s_{all}e_s + c))$, где w — общее количество ФЗ, $O(\texttt{Propagate})$ — сложность функции, локально распространяющей свидетельство, s_{all} — количество вершин в родительском графе, e_s — количество ребер в графе смежности, c — максимальное число конъюнктов в ФЗ. В случае интервальных оценок добавляется сложность поддержания интернальной непротиворечивости.

Допущением этого алгоритма является то, что его корректная работа определяется для ацикличных первичных структур. Поэтому был также представлен алгоритм, позволяющий проверять, представима ли вторичная структура в виде дерева смежности. Алгоритм проверки ацикличности использует только третичную структуру. Он основан на ранее доказанной теореме 1, критерии цикличности, который связывает количество моделей фрагментов знаний в сети с количеством непустых сепараторов и количеством компо-

нент связности сильных сужений в цикличной АБС. Для проверки критерия (равенства) необходимо рассчитать используемые в нем слагаемые. Основную сложность представляет расчет числа компонент связности сильных сужений, для этого была описана функция `GetComponentQuantity`. Работа этой функции опирается на доказанную в статье теорему 2 о принадлежности двух моделей фрагментов знаний к одной компоненте связности сильного сужения. Благодаря этой теореме в родительском графе можно производить определенное в статье распространение цветов от вершин графа к листьям. Распространение цветов вынесено в отдельную процедуру `SpreadColors`, которая используется в `GetComponentQuantity`. После распространения необходимо подсчитать количество различных цветов, где каждый цвет после распространения будет соответствовать одной компоненте связности, при этом цвета могут стать одинаковыми в процессе распространения. Подсчет одинаковых цветов занимает оставшуюся часть функции `GetComponentQuantity`. Таким образом рассчитывается число компонент связности всех сильных сужений, а вместе с тем и условие критерия. Доказана корректность работы этого алгоритма, а также найдена оценка сложности `GetComponentQuantity`, которая лежит в классе $O(sw^2 + e_s s)$, где s — количество непустых сепараторов, w — общее количество ФЗ, а e_s — количество ребер в родительском графе. Стоит отметить, что в результате работы алгоритма находится информация о всех сужениях и сильных сужениях максимального графа смежности на значимые нагрузки, что может использоваться в других алгоритмах, в частности, в описанном в текущей статье алгоритме апостериорного вывода.

Алгоритмы, представленные в работе, были автоматизированы и добавлены в веб-приложение [38]. Данное приложение содержит реализацию не только описанных в статье алгоритмов, но и других связанных с алгебраическими байесовскими сетями. По результатам проведенных исследований была добавлена реализация описанного в статье алгоритма глобального апостериорного вывода, использующего третичную структуру [39], а также алгоритма по проверке ацикличности [40].

В дальнейшем предполагается анализ времени работы данных алгоритмов и сравнение его со временем работы других алгоритмов, решающих те же задачи. Так, алгоритм проверки ацикличности планируется сравнить с алгоритмом, использующим еще одну, новую структуру (четвертичную) [30]. Апостериорный же вывод может проводиться с применением вторичной структуры — механизм распространения виртуальных свидетельств [8]. Основную сложность здесь представляет само построение вторичной структуры [29]. Оценки сложности этих алгоритмов используют понятия, выходящие за рамки данной статьи, но предполагается, что оценки алгоритмов проверки ацикличности сопоставимы по времени, а алгоритм апостериорного вывода, использующий механизм виртуального распространения свидетельств, будет уступать алгоритму, представленному в статье, за счет увеличения времени построения вторичных структур. Сравнение предполагается производить над различными ациклическими АБС. В частности, стоит рассмотреть время работы алгоритмов при различном количестве ФЗ, так как, например, построение вторичных структур будет требовать все больше времени при увеличении количества ФЗ.

Следует добавить, что автоматизация подобных алгоритмов может быть использована в практическом применении алгебраических байесовских сетей, например, для исследования социоинженерных атак [5, 6].

Заключение

Данная статья направлена на решение задачи по расширению и анализу алгоритма глобального апостериорного вывода в АБС, который применяет только третичную структуру [12]. Расширение заключается в дополнении алгоритма до возможности его применения в случае интервальных оценок, ранее использовавшегося только со скалярными оценками. Оно достигнуто за счет поддержания интернальной непротиворечивости. Алгоритм, рассматриваемый в [12] также описан более подробно, то есть представлен в виде псевдокода (процедура `PosterioriInfer`). Доказана корректность работы расширенного алгоритма, а именно показано, что процедура `PosterioriInfer` в случае скалярных оценок назначает оценки вероятностей, совпадающие с результатом пропагации по МГС, а в случае интервальных оценок строит накрывающие оценки, которые бы соответствовали результату пропагации в МГС, если бы распространение виртуального свидетельства давало точные оценки. Помимо этого доказана сложность работы процедуры `PosterioriInfer`.

При этом важным свойством АБС в таком случае является ацикличность, так как только для ацикличных АБС доказана корректность вышеупомянутого алгоритма апостериорного вывода. Таким образом, также изучался вопрос о проверке ацикличности АБС с применением только третичной структуры, подобные алгоритмы ранее не описывались. Предъявленный алгоритм опирается на ранее доказанный критерий ацикличности, представленный в виде равенства (теорема 1). В результате была описана функция `GetComponentQuantity`, которая рассчитывает число компонент связности сильных сужений — одно из слагаемых равенства теоремы, расчет которого представляет наибольшую сложность. Доказана корректность работы этого алгоритма, а также найдена оценка его сложности.

Теоретическая значимость работы заключается в возможном использовании ее результатов в изучении задачи апостериорного вывода в алгебраических байесовских сетях, исследовании третичной структуры как объекта, единственно достаточного для практического применения алгебраических байесовских сетей. Практическая значимость использования результатов заключается в потенциальном ускорении работы алгоритмов апостериорного вывода за счет уменьшения количества формируемых объектов.

В дальнейшем планируется анализ времени работы данных алгоритмов и сравнение его со временем работы других схожих по решаемым задачам алгоритмов, которые направлены на проверку ацикличности [30], а также на глобальный апостериорный вывод, но с использованием вторичных структур [8, 29].

Работа выполнена в рамках проекта по государственному заданию СПБ ФИЦ РАН СПИИРАН № FFZF-2022-0003.

Литература

1. Larrañaga P., Moral S. Probabilistic graphical models in artificial intelligence // Applied Soft Computing. 2011. Vol. 11, no. 2. P. 1511–1528. DOI: 10.1016/j.asoc.2008.01.003.
2. Yang Y., Xu M., Wu W., *et al.* 3D Multiview Basketball Players Detection and Localization Based on Probabilistic Occupancy // 2018 Digital Image Computing: Techniques and Applications (DICTA). IEEE, 2018. P. 1–8. DOI: 10.1109/DICTA.2018.8615798.
3. Masmoudi K., Abid L., Masmoudi A. Credit risk modeling using Bayesian network with a latent variable // Expert Systems with Applications. 2019. Vol. 127. P. 157–166. DOI: 10.1016/j.eswa.2019.03.014.

4. Qiao W., Liu Y., Ma X., Liu Y. Human Factors Analysis for Maritime Accidents Based on a Dynamic Fuzzy Bayesian Network // Risk analysis. 2020. Vol. 40, no. 5. P. 957–980. DOI: 10.1111/risa.13444.

5. Khlobystova A.O., Abramov M.V., Tulupyev A.L. An Approach to Estimating of Criticality of Social Engineering Attacks Traces // International Conference on Information Technologies, Saratov, February 7–8, 2019. Vol. 199. Springer, 2019. P. 446–456. DOI: 10.1007/978-3-030-12072-6_36.

6. Корепанова А.А., Абрамов М.В., Тулупьева Т.В. Идентификация аккаунтов пользователей в социальных сетях «ВКонтакте» и «Одноклассники» // Семнадцатая Национальная конференция по искусственному интеллекту с международным участием. КИИ-2019: сборник научных трудов, Ульяновск, 21–25 октября, 2019. Т. 2. 2019. С. 153–163.

7. Тулупьев А.Л., Николенко С.И., Сироткин А.В. Байесовские сети: логико-вероятностный подход. СПб.: Наука, 2006. 607 с.

8. Тулупьев А.Л. Алгебраические байесовские сети: глобальный логико-вероятностный вывод в деревьях смежности. СПб.: Издательство «Анатолия», 2007. 40 с. Элементы мягких вычислений.

9. Фильченков А.А. Алгоритм построения множества минимальных графов смежности при помощи самоуправляемых клик-собственников // Информатика и автоматизация. 2010. № 14. С. 150–169. DOI: 10.15622/sp.14.9.

10. Фильченков А.А. Алгоритм построения множества минимальных графов смежности при помощи клик-собственников владений // Информатика и автоматизация. 2010. № 15. С. 193–212. DOI: 10.15622/sp.15.10.

11. Фильченков А.А., Тулупьев А.Л. Третичная структура алгебраическое байесовской сети // Информатика и автоматизация. 2011. № 18. С. 164–187. DOI: 10.15622/sp.18.7.

12. Фроленков К.В., Фильченков А.А., Тулупьев А.Л. Апостериорный вывод в третичной полиструктуре алгебраической байесовской сети // Информатика и автоматизация. 2012. № 23. С. 343–356. DOI: 10.15622/sp.23.17.

13. Kabir S., Papadopoulos Y. Applications of Bayesian networks and Petri nets in safety, reliability, and risk assessments: A review // Safety Science. 2019. Vol. 115. P. 154–175. DOI: 10.1016/j.ssci.2019.02.009.

14. Amin M.T., Khan F., Ahmed S., Imtiaz S. A data-driven Bayesian network learning method for process fault diagnosis // Process Safety and Environmental Protection. 2021. Vol. 150. P. 110–122. DOI: 10.1016/j.psep.2021.04.004.

15. Baksh A.-A., Abbassi R., Garaniya V., Khan F. Marine transportation risk assessment using Bayesian Network: Application to Arctic waters // Ocean Engineering. 2018. Vol. 159. P. 422–436. DOI: 10.1016/j.oceaneng.2018.04.024.

16. Cai B., Kong X., Liu Y., *et al.* Application of Bayesian Networks in Reliability Evaluation // IEEE Transactions on Industrial Informatics. 2019. Vol. 15, no. 4. P. 2146–2157. DOI: 10.1109/TII.2018.2858281.

17. Wang Z., Chen C. Fuzzy comprehensive Bayesian network-based safety risk assessment for metro construction projects // Tunnelling and Underground Space Technology. 2017. Vol. 70. P. 330–342. DOI: 10.1016/j.tust.2017.09.012.

18. Tavana M., Abtahi A.-R., Caprio D.D., Poortarigh M. An Artificial Neural Network and Bayesian Network model for liquidity risk assessment in banking // Neurocomputing. 2018. Vol. 275. P. 2525–2554. DOI: 10.1016/j.neucom.2017.11.034.

19. Chaturvedi I., Ragusa E., Gastaldo P., *et al.* Bayesian network based extreme learning machine for subjectivity detection // Journal of the Franklin Institute. 2018. Vol. 355, no. 4. P. 1780–1797. DOI: 10.1016/j.jfranklin.2017.06.007.

20. Ruz G.A., Henríquez P.A., Mascareño A. Sentiment analysis of Twitter data during critical events through Bayesian networks classifiers // Future Generation Computer Systems. 2020. Vol. 106. P. 92–104. DOI: 10.1016/j.future.2020.01.005.

21. Mohammadfam I., Ghasemi F., Kalatpour O., Moghimbeigi A. Constructing a Bayesian network model for improving safety behavior of employees at workplaces // Applied Ergonomics. 2017. Vol. 58. P. 35–47. DOI: 10.1016/j.apergo.2016.05.006.

22. Sierra L.A., Yepes V., García-Segura T., Pellicer E. Bayesian network method for decision-making about the social sustainability of infrastructure projects // Journal of Cleaner Production. 2018. Vol. 176. P. 521–534. DOI: 10.1016/j.jclepro.2017.12.140.

23. McLachlan S., Dube K., Hitman G.A., *et al.* Bayesian networks in healthcare: Distribution by medical condition // Artificial Intelligence in Medicine. 2020. Vol. 107. P. 101912. DOI: 10.1016/j.artmed.2020.101912.

24. Sperotto A., Molina J.-L., Torresan S., *et al.* Reviewing Bayesian Networks potentials for climate change impacts assessment and management: A multi-risk perspective // Journal of Environmental Management. 2017. Vol. 202. P. 320–331. DOI: 10.1016/j.jenvman.2017.07.044.

25. Afenyo M., Khan F., Veitch B., Yang M. Arctic shipping accident scenario analysis using Bayesian Network approach // Ocean Engineering. 2017. Vol. 133. P. 224–230. DOI: 10.1016/j.oceaneng.2017.02.002.

26. Wu J., Zhou R., Xu S., Wu Z. Probabilistic analysis of natural gas pipeline network accident based on Bayesian network // Journal of Loss Prevention in the Process Industries. 2017. Vol. 46. P. 126–136. DOI: 10.1016/j.jlp.2017.01.025.

27. Тулупьев А.Л. Алгебраические байесовские сети: локальный логико-вероятностный вывод. СПб.: Издательство «Анатолия», 2007. 80 с.

28. Тулупьев А.Л. Байесовские сети: логико-вероятностный вывод в циклах. СПб.: Издательство С.-Петербургского университета, 2008. 140 с. Элементы мягких вычислений.

29. Фильченков А.А., Фроленков К.В., Сироткин А.В., Тулупьев А.Л. Система алгоритмов синтеза подмножеств минимальных графов смежности // Информатика и автоматизация. 2013. № 27. С. 200–244. DOI: 10.15622/sp.27.17.

30. Фильченков А.А., Тулупьев А.Л. Алгоритм выявления ацикличности первичной структуры алгебраической байесовской сети по ее четвертичной структуре // Информатика и автоматизация. 2011. № 19. С. 128–145. DOI: 10.15622/sp.19.7.

31. Фильченков А.А., Тулупьев А.Л. Связность и ацикличность первичной структуры алгебраической байесовской сети // Вестник Санкт-Петербургского университета. Математика. Механика. Астрономия. 2013. № 1. С. 110–119.

32. Фильченков А.А., Тулупьев А.Л. Структурный анализ систем минимальных графов смежности // Информатика и автоматизация. 2009. № 11. С. 104–129. DOI: 10.15622/sp.11.6.

33. Сироткин А.В., Тулупьев А.Л. Моделирование знаний и рассуждений в условиях неопределенности: матрично-векторная формализация локального синтеза согласованных оценок истинности // Информатика и автоматизация. 2011. № 18. С. 108–135. DOI: 10.15622/sp.18.5.

34. Фильченков А.А., Тулупьев А.Л. Алгоритм выявления ацикличности первичной структуры алгебраической байесовской сети на основе оценки числа ребер в минимальном графе смежности // Информатика и автоматизация. 2012. № 22. С. 205–223. DOI: 10.15622/sp.22.11.

35. Тулупьев А.Л., Сироткин А.В. Локальный апостериорный вывод в алгебраических байесовских сетях как система матрично-векторных операций // Интегрированные модели и мягкие вычисления в искусственном интеллекте. V-я Международная научно-практическая конференция, 9–12 сентября, 2009. Сборник научных трудов. В 2-х т. Т. 1. СПб.: Наука, 2012. С. 425–434.

36. Aho A., Garey M., Ullman J. The Transitive Reduction of a Directed Graph // SIAM Journal on Computing. 1972. Vol. 1, no. 2. P. 131–137. DOI: 10.1137/0201008.

37. Тулупьев А.Л., Сироткин А.В., Николенко С.И. Байесовские сети доверия: логико-вероятностный вывод в ациклических направленных графах. СПб.: Изд-во Санкт-Петербургского ун-та, 2009. 400 с.

38. Веб-приложение по работе с алгебраическими байесовскими сетями. URL: https://abn.dscs.pro/ (дата обращения: 09.03.2023).

39. Автоматизированные алгоритмы АБС, использующие третичную структуру, в частности — глобальный апостериорный вывод. URL: https://abn.dscs.pro/parent_separators_graph (дата обращения: 09.03.2023).

40. Автоматизированные алгоритмы АБС, работающие с первичной структурой, в частности — проверка ацикличности. URL: https://abn.dscs.pro/primary_structure (дата обращения: 09.03.2023).

Вяткин Артём Андреевич, мл. науч. сотр., Санкт-Петербургский федеральный исследовательский центр Российской академии наук (Санкт-Петербург, Российская Федерация)

Абрамов Максим Викторович, к.т.н., руководитель лаборатории теоретических и междисциплинарных проблем информатики, Санкт-Петербургский федеральный исследовательский центр Российской академии наук (Санкт-Петербург, Российская Федерация), старший научный сотрудник

Харитонов Никита Алексеевич, аспирант, кафедра информатики, Санкт-Петербургский государственный университет (Санкт-Петербург, Российская Федерация)

Тулупьев Александр Львович, д.ф.-м.н., проф., профессор, кафедра бизнес-информатики, Северо-Западный институт управления Российской академии народного хозяйства и государственной службы при Президенте Российской Федерации (Санкт-Петербург, Российская Федерация)

DOI: 10.14529/cmse230104

APPLICATION OF TERTIARY STRUCTURE OF ALGEBRAIC BAYESIAN NETWORK IN THE PROBLEM OF A POSTERIORI INFERENCE

© 2023 A.A. Vyatkin[1], M.V. Abramov[1], N.A. Kharitonov[2], A.L. Tulupyev[3]

[1]*Saint Petersburg Federal Research Center of the Russian Academy of Sciences (14th line 39, Vasilievsky Island, St. Petersburg, 199178 Russia),*

[2]*Saint Petersburg State University (Universitetskaya Emb. 7/9, St. Petersburg, 199034 Russia),*

[3]*North-West Institute of Management of the Russian Presidential Academy of National Economy and Public Administration (Sredniy Ave. 57/43, St. Petersburg, 199034 Russia)*

E-mail: aav@dscs.pro, mva@dscs.pro, nak@dscs.pro, alt@dscs.pro

Received: 09.12.2022

In the theory of algebraic Bayesian networks, there are algorithms that allow to conduct a global posterior inference using secondary structures. At the same time, building secondary structures implies the use of tertiary structure. Consequently, the question about the separate application of the tertiary structure in the problem of a posterior inference arises. This issue has been considered earlier, but only a general description of the algorithm has been given, and only models with scalar estimates of the probability of truth have been taken into account. In this paper, we present an algorithm that extends the aforementioned algorithm to the possibility of using it in the case of interval estimates. In addition, an important property of an algebraic Bayesian network is acyclicality, and the correctness of the above-mentioned algorithms is ensured only for acyclic networks. Therefore, it is also necessary to be able to check the acyclicity of an algebraic Bayesian network using a tertiary structure. The description of this algorithm is also presented in this paper, it is based on the previously proved theorem that relates the number of knowledge pattern models in the network to the number of non-empty separators and the number of strong restriction connectivity components in acyclic algebraic Bayesian network, as well as the theorem proved in this paper that two knowledge pattern models belong to the same strong restriction connectivity component. For all the developed algorithms, the correctness of their performance is proved, and their time complexity estimation is calculated.

Keywords: algebraic Bayesian networks, knowledge pattern, logical and probabilistic inference, tertiary structure, probabilistic graphical models, machine learning.

FOR CITATION

Vyatkin A.A., Abramov M.V., Kharitonov N.A., Tulupyev A.L. Application of Tertiary Structure of Algebraic Bayesian Network in the Problem of a Posteriori Inference. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2023. Vol. 12, no. 1. P. 61–88. (in Russian) DOI: 10.14529/cmse230104.

References

1. Larrañaga P., Moral S. Probabilistic graphical models in artificial intelligence. Applied Soft Computing. 2011. Vol. 11, no. 2. P. 1511–1528. DOI: 10.1016/j.asoc.2008.01.003.
2. Yang Y., Xu M., Wu W., *et al.* 3D Multiview Basketball Players Detection and Localization Based on Probabilistic Occupancy. 2018 Digital Image Computing: Techniques and Applications (DICTA). IEEE. 2018. P. 1–8. DOI: 10.1109/DICTA.2018.8615798.

3. Masmoudi K., Abid L., Masmoudi A. Credit risk modeling using Bayesian network with a latent variable. Expert Systems with Applications. 2019. Vol. 127. P. 157–166. DOI: 10.1016/j.eswa.2019.03.014.

4. Qiao W., Liu Y., Ma X., Liu Y. Human Factors Analysis for Maritime Accidents Based on a Dynamic Fuzzy Bayesian Network. Risk analysis. 2020. Vol. 40, no. 5. P. 957–980. DOI: 10.1111/risa.13444.

5. Khlobystova A.O., Abramov M.V., Tulupyev A.L. An Approach to Estimating of Criticality of Social Engineering Attacks Traces. International Conference on Information Technologies, Saratov, February 7–8, 2019. Vol. 199. Springer. 2019. P. 446–456. DOI: 10.1007/978-3-030-12072-6_36.

6. Korepanova A.A., Abramov M.V., Tulupyeva T.V. Identification of User Accounts in the Social Networks "VKontakte" and "Odnoklassniki". Seventeenth Russian Conference on Artificial Intelligence RCAI-2019: collection of scientific papers, Ulyanovsk, October 21–25, 2019. Vol. 2. 2019. P. 153–163. (in Russian).

7. Tulupyev A.L., Nikolenko S.I., Sirotkin A.V. Bayesian Networks: a Logical and Probabilistic Approach. SPb.: Nauka, 2006. 607 p. (in Russian).

8. Tulupyev A.L. Algebraic Bayesian Networks: Global Logical and Probabilistic Inference in Joint Trees. SPb.: Anatolia Publishing House LLC, 2007. 40 p. Elements of Soft Computing. (in Russian).

9. Filchenkov A.A. Minimal join graph set synthesis self-managed possession cliques algorithm. Informatics and Automation. 2010. No. 14. P. 150–169. (in Russian) DOI: 10.15622/sp.14.9.

10. Filchenkov A.A. Minimal join graph set synthesis proprietor possession cliques algorithm. Informatics and Automation. 2010. No. 15. P. 193–212. (in Russian) DOI: 10.15622/sp.15.10.

11. Filchenkov A.A., Tulupyev A.L. The Algebraic Bayesian Network Tertiary Structure. Informatics and Automation. 2011. No. 18. P. 164–187. (in Russian) DOI: 10.15622/sp.18.7.

12. Frolenkov K.V., Filchenkov A.A., Tulupyev A.L. Posteriori inference in tertiary polystructure of an algebraic Bayesian network. Informatics and Automation. 2012. No. 23. P. 343–356. (in Russian) DOI: 10.15622/sp.23.17.

13. Kabir S., Papadopoulos Y. Applications of Bayesian networks and Petri nets in safety, reliability, and risk assessments: A review. Safety Science. 2019. Vol. 115. P. 154–175. DOI: 10.1016/j.ssci.2019.02.009.

14. Amin M.T., Khan F., Ahmed S., Imtiaz S. A data-driven Bayesian network learning method for process fault diagnosis. Process Safety and Environmental Protection. 2021. Vol. 150. P. 110–122. DOI: 10.1016/j.psep.2021.04.004.

15. Baksh A.-A., Abbassi R., Garaniya V., Khan F. Marine transportation risk assessment using Bayesian Network: Application to Arctic waters. Ocean Engineering. 2018. Vol. 159. P. 422–436. DOI: 10.1016/j.oceaneng.2018.04.024.

16. Cai B., Kong X., Liu Y., *et al.* Application of Bayesian Networks in Reliability Evaluation. IEEE Transactions on Industrial Informatics. 2019. Vol. 15, no. 4. P. 2146–2157. DOI: 10.1109/TII.2018.2858281.

17. Wang Z., Chen C. Fuzzy comprehensive Bayesian network-based safety risk assessment for metro construction projects. Tunnelling and Underground Space Technology. 2017. Vol. 70. P. 330–342. DOI: 10.1016/j.tust.2017.09.012.

18. Tavana M., Abtahi A.-R., Caprio D.D., Poortarigh M. An Artificial Neural Network and Bayesian Network model for liquidity risk assessment in banking. Neurocomputing. 2018. Vol. 275. P. 2525–2554. DOI: 10.1016/j.neucom.2017.11.034.

19. Chaturvedi I., Ragusa E., Gastaldo P., *et al.* Bayesian network based extreme learning machine for subjectivity detection. Journal of the Franklin Institute. 2018. Vol. 355, no. 4. P. 1780–1797. DOI: 10.1016/j.jfranklin.2017.06.007.

20. Ruz G.A., Henríquez P.A., Mascareño A. Sentiment analysis of Twitter data during critical events through Bayesian networks classifiers. Future Generation Computer Systems. 2020. Vol. 106. P. 92–104. DOI: 10.1016/j.future.2020.01.005.

21. Mohammadfam I., Ghasemi F., Kalatpour O., Moghimbeigi A. Constructing a Bayesian network model for improving safety behavior of employees at workplaces. Applied Ergonomics. 2017. Vol. 58. P. 35–47. DOI: 10.1016/j.apergo.2016.05.006.

22. Sierra L.A., Yepes V., García-Segura T., Pellicer E. Bayesian network method for decision-making about the social sustainability of infrastructure projects. Journal of Cleaner Production. 2018. Vol. 176. P. 521–534. DOI: 10.1016/j.jclepro.2017.12.140.

23. McLachlan S., Dube K., Hitman G.A., *et al.* Bayesian networks in healthcare: Distribution by medical condition. Artificial Intelligence in Medicine. 2020. Vol. 107. P. 101912. DOI: 10.1016/j.artmed.2020.101912.

24. Sperotto A., Molina J.-L., Torresan S., *et al.* Reviewing Bayesian Networks potentials for climate change impacts assessment and management: A multi-risk perspective. Journal of Environmental Management. 2017. Vol. 202. P. 320–331. DOI: 10.1016/j.jenvman.2017.07.044.

25. Afenyo M., Khan F., Veitch B., Yang M. Arctic shipping accident scenario analysis using Bayesian Network approach. Ocean Engineering. 2017. Vol. 133. P. 224–230. DOI: 10.1016/j.oceaneng.2017.02.002.

26. Wu J., Zhou R., Xu S., Wu Z. Probabilistic analysis of natural gas pipeline network accident based on Bayesian network. Journal of Loss Prevention in the Process Industries. 2017. Vol. 46. P. 126–136. DOI: 10.1016/j.jlp.2017.01.025.

27. Tulupyev A.L. Algebraic Bayesian Networks: Local Logical and Probabilistic Inference. SPb.: Anatolia Publishing House LLC, 2007. 80 p. (in Russian).

28. Tulupyev A.L. Bayesian Networks: Logic-probabilistic Inference in Cycles. SPb.: St. Petersburg University Press, 2008. 140 p. Elements of Soft Computing. (in Russian).

29. Filchenkov A.A., Frolenkov K.V., Sirotkin A.V., Tulupyev A.L. Minimal join graph subsets synthesis system. Informatics and Automation. 2013. No. 27. P. 200–244. (in Russian) DOI: 10.15622/sp.27.17.

30. Filchenkov A.A., Tulupyev A.L. Algorithm for detection of algebraic Bayesian network primary structure acyclicity based on its quaternary structure. Informatics and Automation. 2011. No. 19. P. 128–145. (in Russian) DOI: 10.15622/sp.19.7.

31. Filchenkov A.A., Tulupyev A.L. Connectivity and Acyclicity of the Primary Structure of an Algebraic Bayesian Network. Vestnik of Saint Petersburg University. Mathematics. Mechanics. Astronomy. 2013. No. 1. P. 110–119. (in Russian).

32. Filchenkov A.A., Tulupyev A.L. Minimal joint graph structure synthesis. Informatics and Automation. 2009. No. 11. P. 104–129. (in Russian) DOI: 10.15622/sp.11.6.

33. Sirotkin A.V., Tulupyev A.L. Knowledge and reasoning with uncertainty modeling: matrix-and-vector calculus for local reconciliation of truth estimates. Informatics and Automation. 2011. No. 18. P. 108–135. (in Russian) DOI: 10.15622/sp.18.5.

34. Filchenkov A.A., Tulupyev A.L. Algorithm for Detection Algebraic Bayesian Network Primary Structure Acyclicity Based on Number of Minimal Join Graph Edges Estimating. Informatics and Automation. 2012. No. 22. P. 205–223. (in Russian) DOI: 10.15622/sp.22.11.

35. Tulupyev A.L., Sirotkin A.V. Local Posterior Inference in Algebraic Bayesian Networks as a System of Matrix-and-vector Operations. Integrated Models and Soft Computing in Artificial Intelligence. V-th International Scientific and Practical Conference, September 9–12, 2009. Collection of scientific works. In 2 vols. Vol. 1. SPb.: Nauka, 2012. P. 425–434. (in Russian).

36. Aho A., Garey M., Ullman J. The Transitive Reduction of a Directed Graph. SIAM Journal on Computing. 1972. Vol. 1, no. 2. P. 131–137. DOI: 10.1137/0201008.

37. Tulupyev A.L., Sirotkin A.V., Nikolenko S.I. Bayesian Belief Networks: Logical and Probabilistic Inference in Acyclic Directed Graphs. SPb.: St. Petersburg University Press, 2009. 400 p. (in Russian).

38. Web application for algebraic Bayesian networks. URL: https://abn.dscs.pro/ (accessed: 09.03.2023).

39. Automated ABN algorithms using tertiary structure, in particular – global a posterior inference. URL: https://abn.dscs.pro/parent_separators_graph (accessed: 09.03.2023).

40. Automated ABN algorithms working with the primary structure, in particular – check acyclicity. URL: https://abn.dscs.pro/primary_structure (accessed: 09.03.2023).

УДК 004.421, 519.688 DOI: 10.14529/cmse230105

ВЫДЕЛЕНИЕ ЯВНОГО УРОВНЯ РЕАЛИЗАЦИИ АЛГОРИТМОВ ДЛЯ ИСПОЛЬЗОВАНИЯ В ПРОЕКТЕ ALGO500

© 2023 А.С. Антонов

Научно-иссследовательский вычислительный центр
Московского государственного университета имени М.В. Ломоносова
(119234 Москва, ул. Ленинские горы, д. 1, стр. 4)
E-mail: asa@parallel.ru

Поступила в редакцию: 03.12.2022

Исследование и описание свойств алгоритмов крайне важно для их эффективной реализации на различных типах целевых программно-аппаратных платформ. Этой актуальной задаче посвящен проект создания Открытой энциклопедии свойств алгоритмов AlgoWiki, начатый в Московском государственном университете имени М.В. Ломоносова в 2014 году. В рамках проекта была предложена единая универсальная схема описания свойств алгоритмов, в которой особое внимание уделялось свойствам, связанным с параллелизмом. Множество описанных по данной схеме алгоритмов послужило основой описания структуры предметной области в рамках иерархической схемы «Задача—Метод—Алгоритм–Реализация». Однако для дальнейшего развития проекта AlgoWiki потребовалось выделить реализации алгоритмов, ранее включенные в описания свойств алгоритмов, в отдельную сущность. В данной статье изложена схема описания свойств реализаций алгоритмов, также как и модификация изначальной схемы описания свойств самих алгоритмов. Преобразование описаний в энциклопедии AlgoWiki по данной схеме было выполнено для всех страниц проекта, и оно позволяет как более качественно описывать свойства реализаций алгоритмов, так и интегрировать проект энциклопедии AlgoWiki с проектом Algo500, реализующим масштабируемую цифровую платформу для совместного анализа свойств алгоритмов и компьютерных архитектур.

Ключевые слова: Algo500, AlgoWiki, задача, метод, алгоритм, реализация, суперкомпьютер, рейтинг.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Антонов А.С. Выделение явного уровня реализации алгоритмов для использования в проекте Algo500 // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2023. Т. 12, № 1. С. 89–100. DOI: 10.14529/cmse230105.

Введение

Проект создания Открытой энциклопедии свойств алгоритмов AlgoWiki [1, 2] стартовал в Московском государственном университете имени М.В. Ломоносова в 2014 году. Его основной целью было создание доступной платформы для описания свойств вычислительных алгоритмов [3] силами формируемого вычислительного сообщества. Такая платформа с использованием wiki-технологий была создана на базе движка MediaWiki [4], разработанного изначально для проекта Wikipedia [5]. С использованием данных технологий был создан сайт энциклопедии AlgoWiki [6].

На первых этапах создания Открытой энциклопедии свойств алгоритмов AlgoWiki был реализован ряд основных разделов сайта:

- классификация алгоритмов — основной раздел энциклопедии AlgoWiki, в котором упорядочиваются все имеющиеся в проекте страницы описаний;
- структура описания свойств алгоритмов — разработанная универсальная схема, по которой предлагается описывать свойства и структуру каждого вычислительного алгоритма;

- руководства по заполнению разделов описания — пошаговые инструкции по заполнению разделов описаний алгоритмов;
- готовность статей — механизм, выдающий списки статей, размеченных авторами по признакам «Начатые статьи», «Статьи в работе», «Законченные статьи»;
- глоссарий — раздел для описания используемых в проекте терминов;
- помощь — раздел со справочными материалами.

Целью данной работы является разработка новой схемы описания свойств реализаций алгоритмов, описанных в энциклопедии AlgoWiki. Использование предлагаемой схемы позволяет закончить реализацию структуры иерархического представления предметной области. Помимо этого, выделение реализаций алгоритмов необходимо для их использования в рамках проекта создания масштабируемой цифровой платформы Algo500 [7, 8], в которой осуществляется интеграция энциклопедии AlgoWiki с идеями, используемыми в известных рейтингах высокопроизводительных вычислительных систем.

Статья организована следующим образом. В разделе 1 рассмотрена ранее разработанная в рамках энциклопедии AlgoWiki единая универсальная схема описания свойств алгоритмов. В разделе 2 описано иерархическое представление предметной области в виде цепочек «Задача–Метод–Алгоритм–Реализация». Раздел 3 посвящен выделению явного уровня реализации алгоритмов в энциклопедии AlgoWiki. В разделе 4 предложена новая структура описания выделенных реализаций алгоритмов. Раздел 5 описывает направления развития энциклопедии AlgoWiki в рамках проекта Algo500. В заключении приводится краткая сводка результатов, полученных в работе, и указаны направления дальнейших исследований.

1. Схема описания свойств алгоритмов

Изначально была предложена единая универсальная схема описания свойств алгоритмов, состоявшая из двух частей. В первую часть входили описания машинно-независимых свойств алгоритмов, а во вторую — свойства, которые могут различаться при реализации на различных программно-аппаратных платформах. На рис. 1 приведена получившаяся схема, по которой предлагалось описывать все вычислительные алгоритмы в AlgoWiki.

Данная структура была использована для описания большого количества алгоритмов из самых разных областей науки. Отдельный акцент во всех описаниях делается на свойствах, связанных с параллелизмом. Были предложены и широко использованы технологии описания отдельных разделов таких описаний. В частности, для изображения информационных графов [9], являющихся основой анализа параллельных свойств алгоритмов, в разделе описания 1.7 была использована разработанная система интерактивной трехмерной визуализации информационных графов AlgoView [10, 11, 12]. Данная система позволяет отобразить параллельную структуру анализируемого алгоритма, оценить ресурс параллелизма, пошагово посмотреть ярусно-параллельную форму алгоритма, получить некоторые его численные характеристики и т.д. Пример отображения информационного графа при помощи системы AlgoView для алгоритма Холецкого [13] в рамках энциклопедии AlgoWiki приведен на рис. 2.

2. Иерархическое представление предметной области

Со временем усилиями привлеченных исследователей в Открытой энциклопедии свойств алгоритмов AlgoWiki накопилось большое количество страниц описаний алгоритмов в разной степени готовности. На множестве описаний алгоритмов требовалось выпол-

```
1 Свойства и структура алгоритмов
    1.1 Общее описание алгоритма
    1.2 Математическое описание алгоритма
    1.3 Вычислительное ядро алгоритма
    1.4 Макроструктура алгоритма
    1.5 Схема реализации последовательного алгоритма
    1.6 Последовательная сложность алгоритма
    1.7 Информационный граф
    1.8 Ресурс параллелизма алгоритма
    1.9 Входные и выходные данные алгоритма
    1.10 Свойства алгоритма
2 Программная реализация алгоритма
    2.1 Особенности реализации последовательного алгоритма
    2.2 Локальность данных и вычислений
    2.3 Возможные способы и особенности параллельной реализации алгоритма
    2.4 Масштабируемость алгоритма и его реализации
    2.5 Динамические характеристики и эффективность реализации алгоритма
    2.6 Выводы для классов архитектур
    2.7 Существующие реализации алгоритма
3 Литература
```

Рис. 1. Структура описания свойств алгоритма

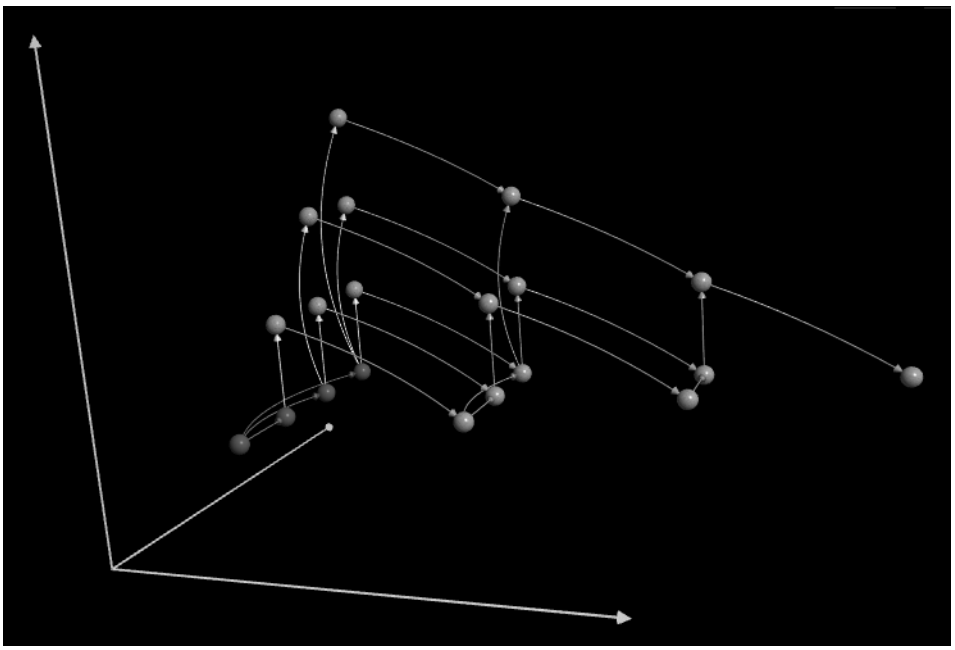


Рис. 2. Информационный граф алгоритма Холецкого

нить какое-то упорядочивание. Поиск существующих классификаций вычислительных алгоритмов не дал существенных результатов, поэтому в рамках проекта стали возникать первые версии классификации алгоритмов, в которых авторы разносили свои описания по тематическим разделам из разных областей науки. Однако структура такой страницы оставалась неопределенной, а отнесение статей к тем или иным разделам оставалось достаточно произвольным.

Достаточно очевидно, что вычислительные алгоритмы нужны не сами по себе, а для решения задач, возникающих в различных областях науки и промышленности. В то же время многие практические задачи можно решать, применяя различные методы. В свою очередь, каждый из методов обладает своими свойствами, и в определенных условиях может быть выгодно использовать один из них. Такие условия может определять целевая программно-аппаратная среда. Для реализации методов могут использоваться те или иные алгоритмы, а программирование алгоритмов с использованием технологий параллельного программирования приводит к возникновению программных реализаций, которые выполняются в операционной среде целевого суперкомпьютера.

Таким образом, мы получаем широко известную схему отображения вычислительных задач на вычислительные системы, которую было решено реализовать в рамках Открытой энциклопедии свойств алгоритмов AlgoWiki. В энциклопедию были добавлены страницы, описывающие решаемые задачи и используемые методы, и возникло иерархическое представление предметной области в виде цепочек «Задача–Метод–Алгоритм–Реализация» [14, 15]. На рис. 3 показаны возможные соотношения между этими описаниями. Так, для описания исследуемой задачи могут использоваться страницы описаний других задач, может осуществляться выбор метода ее решения или же выбираться сразу готовый алгоритм реализации. Описания метода могут базироваться на описаниях других методов, или же для него может выбираться алгоритм реализации. Для описания алгоритма в качестве составных частей могут использоваться другие алгоритмы, после же фиксации алгоритмической стороны и с учетом особенностей целевой программно-аппаратной платформы выполнятся программная реализация.

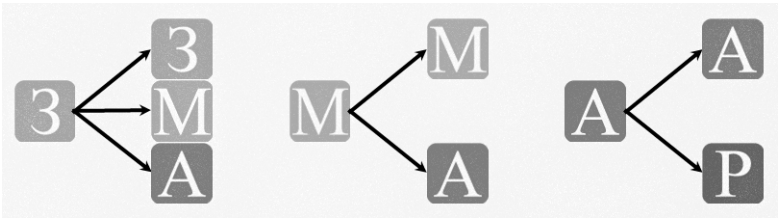


Рис. 3. Возможные связи на странице классификации (З — задача, М — метод, А — алгоритм, Р — реализация)

Для поддержания данной иерархической структуры выполнено отступление от концепции wiki-представления информации — работа со страницей классификации алгоритмов ведется через систему меню, разработанных в рамках проекта AlgoWiki [16]. Посредством этих меню авторизованный пользователь может добавлять в классификацию алгоритмов новые страницы, соответствующие задачам, методам, алгоритмам и реализациям, придерживаясь ограничений, приведенных на рис. 3. При этом работа с самими описаниями задач, методов, алгоритмов и реализаций по-прежнему ведется в рамках wiki-концепции, что позволяет поддерживать коллективную разработку силами вычислительного сообщества.

3. Выделение явного уровня реализации алгоритмов

Описанная иерархическая схема описания предметной области хорошо сочеталась с предложенной ранее схемой описания свойств алгоритмов (рис. 1) за одним исключением: уровень реализации алгоритмов не был выделен явно, а оставался частью описания соответствующего алгоритма. Получалось, что множество различных реализаций одного алгорит-

ма (зачастую предназначенных для различных программно-аппаратных платформ) описывалось в рамках одного раздела. Динамические характеристики из второй части описания алгоритма [17] (локальность [18], масштабируемость [19] и другие) существенно отличаются для каждой реализации, поэтому должны описываться раздельно. Все это приводило к крайней перегруженности описаний алгоритмов, особенно тех, у которых описывалось большое количество различных реализаций (для некоторых алгоритмов авторы приводили десятки разных реализаций).

Поэтому далее в рамках работ по проекту было решено провести выделение описаний реализации алгоритмов в отдельную сущность. Для этого была изменена вторая часть описаний алгоритмов, из которой убрали все, что относится не к алгоритму как таковому, а к конкретным его реализациям. Получившаяся после этого структура второй части описания свойств алгоритмов приведена на рис. 4.

```
2 Программная реализация алгоритма
    2.1 Особенности реализации последовательного алгоритма
    2.2 Возможные способы и особенности параллельной реализации алгоритма
    2.3 Результаты прогонов
    2.4 Выводы для классов архитектур
```

Рис. 4. Новая структура второй части описания свойств алгоритма

Разделы 2.1, 2.2 и 2.4 остались во второй части описания свойств алгоритма, поскольку в них суммируются выводы по различным программным реализациям. Кроме того, в описание был добавлен раздел 2.3, в котором также суммируются данные с результатов прогонов различных реализаций данного алгоритма на различных программно-аппаратных платформах (это делается в рамках реализации проекта создания цифровой платформы Algo500 [7, 8]).

4. Описание свойств реализации алгоритма

Теперь в рамках Открытой энциклопедии свойств алгоритмов AlgoWiki для каждой реализации каждого алгоритма предлагается выделять отдельную страницу, на которой описываются ее основные свойства. Предложенная структура описания такой страницы приведена на рис. 5.

```
1 Ссылки
2 Локальность данных и вычислений
    2.1 Локальность реализации алгоритма
        2.1.1 Структура обращений в память и качественная оценка локальности
        2.1.2 Количественная оценка локальности
3 Масштабируемость алгоритма и его реализации
    3.1 Масштабируемость алгоритма
    3.2 Масштабируемость реализации алгоритма
4 Динамические характеристики и эффективность реализации алгоритма
5 Результаты прогонов
```

Рис. 5. Структура описания свойств реализации алгоритма

В разделе 1 описания реализации алгоритма приводятся ссылки, позволяющие найти данную реализацию и информацию о ней. Разделы 2–4 перенесены сюда со страниц описаний свойств алгоритмов, поскольку содержат фактически описания свойств конкретных реализаций. Также добавляется новый раздел 5, в котором приводятся результаты прогонов данной программной реализации на различных программно-аппаратных платформах (это делается в рамках реализации проекта создания цифровой платформы Algo500 [7, 8]). В этом разделе считываются данные из базы проекта Algo500 и отображаются в виде таблиц или графиков.

В итоге все реализации всех описанных алгоритмов из Открытой энциклопедии свойств алгоритмов AlgoWiki были выделены в отдельные страницы с описанием по схеме, приведенной на рис. 5. Это позволило завершить процесс описания структуры предметной области в виде цепочек «Задача–Метод–Алгоритм–Реализация». Фрагмент получившейся страницы «Классификация алгоритмов» приведен на рис. 6.

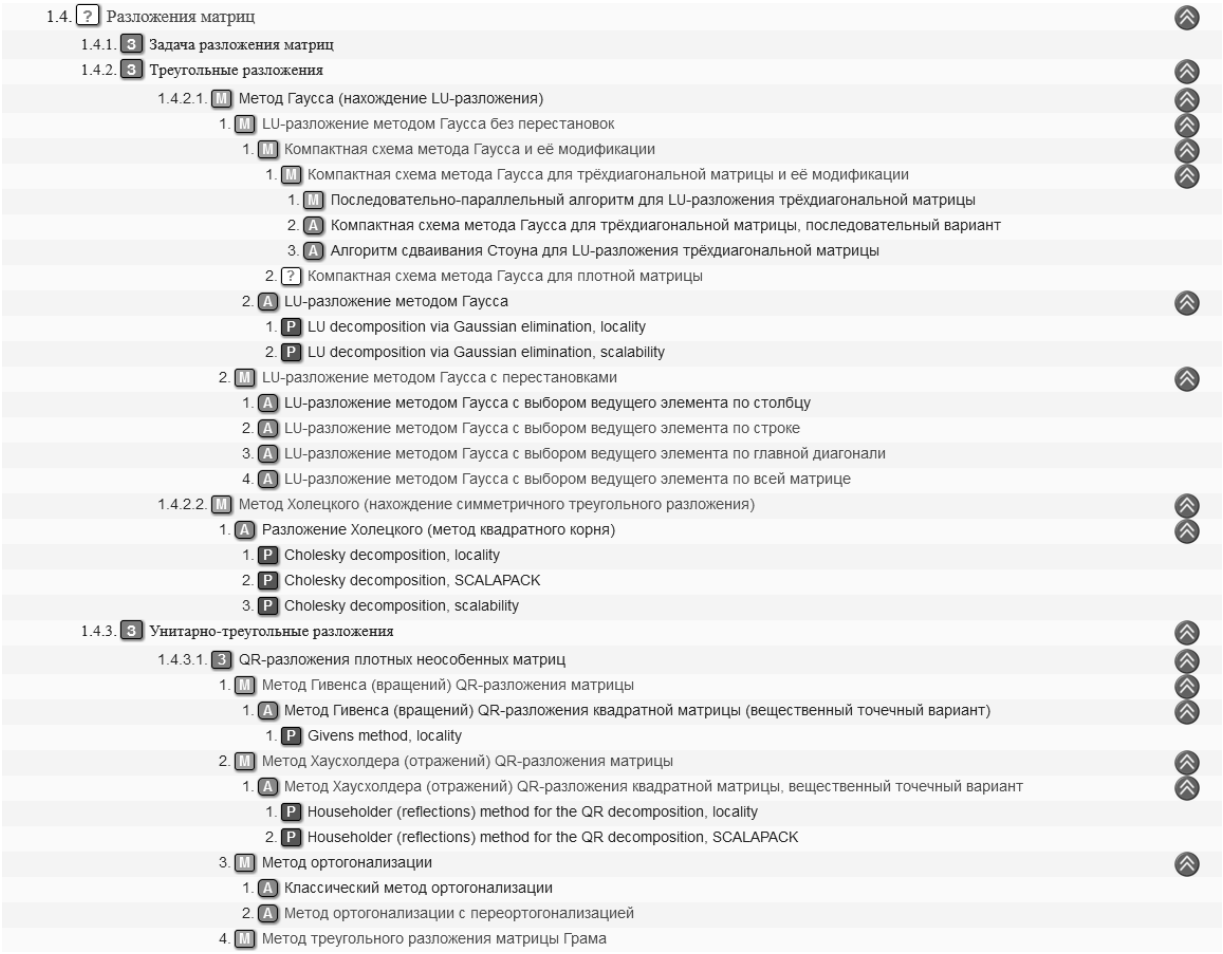


Рис. 6. Фрагмент страницы «Классификация алгоритмов»

Так, на рис. 6 можно отследить цепочку описаний для алгоритма Холецкого, информационный граф которого был приведен на рис. 2. В разделе «Разложения матриц» (для которого текст пока не написан, что вполне нормально для живого wiki-проекта) выделена отдельная задача «Треугольные разложения», в которой подробно описана страница «Метод Холецкого (нахождение симметричного треугольного разложения)». Для указанного метода описан алгоритм реализации «Разложение Холецкого (метод квадратного корня)»,

а для него отдельными страницами приводятся три реализации: последовательная реализация, использованная для изучения локальности данных, параллельная реализация из пакета SCALAPACK библиотеки Intel MKL (метод pdpotrf), использованная для исследования динамических характеристик и эффективности реализации алгоритма, и написанная одним из авторов параллельная реализация на языке Си, использованная для исследования масштабируемости реализации алгоритма. Все эти три реализации существенно отличаются и обладают различными свойствами, поэтому описания их в рамках одной страницы было не вполне корректно. Кроме того, после выделения отдельных страниц реализаций их можно использовать в рамках проекта Algo500.

Подобное преобразование страницы классификации алгоритмов было выполнено для явного выделений описаний всех реализаций всех алгоритмов, описанных в проекте.

5. Проект Algo500

В настоящее время в рамках проекта реализуется масштабируемая цифровая платформа Algo500 [7, 8], которая обеспечивает следующие основные функции:

- объединение данных о любых алгоритмах и архитектурах компьютеров;
- подход с единых позиций к анализу свойств любого алгоритма применительно к особенностям любой архитектуры;
- возможность вычислительному сообществу дополнять и уточнять базу алгоритмов, их реализаций, вносить данные об их выполнении на различных вычислительных системах;
- формирование по запросу произвольных индивидуальных рейтинговых списков.

В рамках проекта Algo500 производится интеграция четырех основных программных компонент:

1. Открытая энциклопедия свойств алгоритмов AlgoWiki.
2. База описаний компьютерных систем CompZoo.
3. Репозиторий исходных кодов и результатов запусков PerfData.
4. Система построения настраиваемых рейтингов RatingLists.

Из энциклопедии AlgoWiki можно получить доступ к любым задачам, методам, алгоритмам и их реализациям. При этом, зафиксировав нужный уровень, можно делать выборки данных из других компонент проекта, включающие информацию для данного уровня и всех последователей в иерархическом представлении предметной области.

В базу данных CompZoo заносятся структурированные описания архитектур суперкомпьютеров, на которых проводятся эксперименты, с указанием их характеристик, оказывающих наиболее существенное влияние на время выполнения, производительность, эффективность и другие динамические характеристики. Далее можно делать выборки данных из других компонент проекта, включающие информацию для суперкомпьютеров с определенными пользователям характеристиками.

В компоненту PerfData заносятся программные реализации алгоритмов, конфигурационные файлы для конкретных вычислительных систем, сведения о использованной совокупности вычислительных узлов суперкомпьютера, данные о входных параметрах и полученных результатах прогона конкретной программной реализации. Далее пользователь может делать выборки данных из других компонент проекта, включающие информацию для прогонов с определенными параметрами или полученными результатами.

В качестве одного из основных результатов цифровая платформа Algo500 позволяет в рамках компоненты RatingLists строить рейтинговые списки [20] по любым описанным реализациям любых алгоритмов из энциклопедии AlgoWiki. Такая система рейтингов не только включает в себя самые известные рейтинги в данной области (Top500 [21] на основе теста Linpack [22], Graph500 [23, 24], HPCG [25, 26] и другие), но и позволяет построить полноценную систему рейтингов с большими возможностями по представлению информации, построению различных выборок и получению аналитических оценок.

Заключение

В данной статье описано развитие проекта создания Открытой энциклопедии свойств алгоритмов AlgoWiki. Очередным важным шагом в выполнении данного проекта явилось выделение явного уровня реализаций алгоритмов. Была предложена структура описания реализаций алгоритмов, а также модификация структуры описания самих алгоритмов. Согласно этим структурам было выполнено преобразование всех описаний алгоритмов в рамках энциклопедии AlgoWiki и соответствующим образом модифицирована страница классификации алгоритмов. Это позволило завершить создание иерархического представления предметной области в виде цепочек «Задача–Метод–Алгоритм–Реализация».

Дальнейшим развитием проекта является его полноценная интеграция с механизмами, разработанными в рамках реализации масштабируемой цифровой платформы Algo500, что предоставляет возможность совместного анализа страниц энциклопедии AlgoWiki с описаниями программно-аппаратных платформ, используемых для высокопроизводительных вычислений.

Результаты получены в Московском государственном университете имени М.В. Ломоносова при финансовой поддержке РНФ (договор № 20–11–20194). Работа выполнена с использованием оборудования Центра коллективного пользования сверхвысокопроизводительными вычислительными ресурсами МГУ имени М.В. Ломоносова [27].

Литература

1. Voevodin V., Antonov A., Dongarra J. AlgoWiki: an Open Encyclopedia of Parallel Algorithmic Features // Supercomput. Front. and Innov. 2015. Vol. 1, no. 2. P. 4–18. DOI: 10.14529/jsfi150101.
2. Воеводин В. Открытая энциклопедия свойств алгоритмов AlgoWiki: от мобильных платформ до экзафлопсных суперкомпьютерных систем // Вычислительные методы и программирование 2015. Т. 16, № 1. С. 99–111. DOI: 10.26089/NumMet.v16r111.
3. Voevodin V., Antonov A., Dongarra J. Why is it hard to describe properties of algorithms? // Procedia Computer Science. 2016. Vol. 101. P. 4–7. DOI: 10.1016/j.procs.2016.11.002.
4. MediaWiki. URL: `https://www.mediawiki.org` (дата обращения: 01.12.2022).
5. Wikipedia. URL: `https://wikipedia.org` (дата обращения: 01.12.2022).
6. Открытая энциклопедия свойств алгоритмов. URL: `http://algowiki-project.org` (дата обращения: 01.12.2022).
7. Antonov A., Nikitenko D., Voevodin Vl. Algo500 — a New Approach to the Joint Analysis of Algorithms and Computers // Lobachevskii Journal of Mathematics. 2020. Vol. 41, no. 8. P. 1435–1443. DOI: 10.1134/S1995080220080041.

8. Antonov A.S., Maier R.V. Development and Implementation of the Algo500 Scalable Digital Platform Architecture // Lobachevskii J Math. 2022. Vol. 43. P. 837–847. DOI: 10.1134/S1995080222070058.

9. Воеводин В., Воеводин Вл. Параллельные вычисления. Санкт-Петербург: БХВ-Петербург, 2002. 608 с.

10. Antonov A.S., Volkov N.I. An AlgoView Web-visualization System for the AlgoWiki Project // Communications in Computer and Information Science. 2017. Vol. 753. P. 3–13. DOI: 10.1007/978-3-319-67035-5_1.

11. Antonov A., Volkov N. Interactive 3D Representation as a Method of Investigating Information Graph Features // Communications in Computer and Information Science. 2018. Vol. 965. P. 587–598. DOI: 10.1007/978-3-030-05807-4_50.

12. Antonov A.S., Volkov N.I. Information Graph Visualization Using AlgoView Software Tool // Lobachevskii J Math. 2020. Vol. 41, no. 6. P. 1427–1434. DOI: 10.1134/S199508022008003X.

13. Cholesky A.-L. Sur la rèsolution numérique des systèmes déquations linéaires // La SABIX, Bulletins déja publiés, Sommaire du bulletin. 2005. No. 39. P. 81–95.

14. Antonov A., Frolov A., Konshin I., Voevodin Vl. Hierarchical Domain Representation in the AlgoWiki Encyclopedia: From Problems to Implementations // Communications in Computer and Information Science. 2018. Vol. 910. P. 3–15. DOI: 10.1007/978-3-319-99673-8_1.

15. Popov A., Nikitenko D., Antonov A., Voevodin Vl. Formal model of problems, methods, algorithms and implementations in the advancing AlgoWiki open encyclopedia // CEUR Workshop Proc. 2018. Vol. 2281. P. 1–11.

16. Antonov A.S., Maier R.V. A New Representation of Algorithmic Approaches in the AlgoWiki Encylopedia // Lobachevskii J Math. 2021. Vol. 42, no. 7. P. 1483–1491. DOI: 10.1134/S1995080221070039.

17. Antonov A., Voevodin Vad., Voevodin Vl., Teplov A. A Study of the Dynamic Characteristics of Software Implementation as an Essential Part for a Universal Description of Algorithm Properties // 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing Proceedings, February 17–19, 2016. P. 359–363. DOI: 10.1109/PDP.2016.24.

18. Воеводин В.В., Швец П.А. Метод покрытий для оценки локальности использования данных в программах // Вестник Уфимского государственного авиационного технического университета. 2014. Т. 18, № 1(62). С. 224–229.

19. Antonov A., Teplov A. Generalized approach to scalability analysis of parallel applications // Lecture Notes in Computer Science. 2016. Vol. 10049. P. 291–304. DOI: 10.1007/978-3-319-49956-7_23.

20. Antonov A., Dongarra J., Voevodin V. AlgoWiki Project as an Extension of the Top500 Methodology // Supercomputing Frontiers and Innovations. 2018. Vol. 5, no. 1. P. 4–10. DOI: 10.14529/jsfi180101.

21. Home - | TOP500. URL: `https://top500.org` (дата обращения: 01.12.2022).

22. Dongarra J.J., Bunch J.R., Moler G.B., Stewart G.W. LINPACK Users' Guide. Society for Industrial and Applied Mathematics, 1979–1993.

23. Graph 500 | large-scale benchmarks. URL: `https://graph500.org` (дата обращения: 01.12.2022).

24. Murphy R.C., Wheeler K.B., Barrett B.W., Ang J.A. Introducing the Graph 500. Cray User's Group (CUG). May 5, 2010. Vol. 19. P. 45–74.

25. HPCG Benchmark. URL: `https://www.hpcg-benchmark.org` (дата обращения: 01.12.2022).

26. Heroux M., Dongarra J. Toward a New Metric for Ranking High Performance Computing Systems. UTK EECS Tech Report and Sandia National Labs Report SAND2013-4744, June 2013.

27. Voevodin V., Antonov A., Nikitenko D., *et al.* Supercomputer Lomonosov-2: Large Scale, Deep Monitoring and Fine Analytics for the User Community // Supercomputing Frontiers and Innovations. 2019. Vol. 6, no. 2. P. 4–11. DOI: 10.14529/jsfi190201.

Антонов Александр Сергеевич, к.ф.-м.н., ведущий научный сотрудник, Научно-исследовательский вычислительный центр, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация)

DOI: 10.14529/cmse230105

EXTRACTION OF AN EXPLICIT LEVEL OF ALGORITHM IMPLEMENTATION FOR USE IN THE ALGO500 PROJECT

Research Computing Center, Lomonosov Moscow State Universisty
(GSP-1, Leninskie Gory 1, building 4, Moscow, 119234 Russia)
E-mail: asa@parallel.ru
Received: 03.12.2022

The study and description of the algorithm properties is extremely important for their effective implementation on various types of target software and hardware platforms. This topical task is the subject of a project to create an AlgoWiki Open encyclopedia of parallel algorithmic features, launched at Lomonosov Moscow State University in 2014. Within the framework of the project, a unified universal scheme for describing the algorithm properties was proposed, in which special attention to the properties associated with parallelism was paid. The set of algorithms described according to this scheme served as the basis for describing the structure of the subject area within the framework of the "Problem–Method–Algorithm–Implementation" hierarchical scheme. However, for the further development of the AlgoWiki project, it was necessary to separate the implementations of the algorithms that were previously included in the descriptions of the properties of the algorithms. This paper presents a scheme for describing the properties of algorithm implementations, as well as a modification of the original scheme for describing the properties of the algorithms themselves. The transformation of descriptions in the AlgoWiki encyclopedia according to this scheme was performed for all pages of the project, and it allows not only to better describe the properties of algorithm implementations, but also to integrate the AlgoWiki encyclopedia project with the Algo500 project, which implements a scalable digital platform for joint analysis of the properties of algorithms and computer architectures.

Keywords: Algo500, AlgoWiki, problem, method, algorithm, implementation, supercomputer, rating.

FOR CITATION

Antonov A.S. Extraction of an Explicit Level of Algorithms Implementation for Use in the Algo500 Project. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2023. Vol. 12, no. 1. P. 89–100. (in Russian) DOI: 10.14529/cmse230105.

References

1. Voevodin V., Antonov A., Dongarra J. AlgoWiki: an Open Encyclopedia of Parallel Algorithmic Features. Supercomput. Front. and Innov. 2015. Vol. 1, no. 2. P. 4–18. DOI: 10.14529/jsfi150101.

2. Voevodin V. An AlgoWiki open encyclopedia of algorithmic features: from mobile to extreme scale. Numer. methods and program. 2015. Vol. 16, no. 1. P. 99–111. (in Russian) DOI: 10.26089/NumMet.v16r111.

3. Voevodin V., Antonov A., Dongarra J. Why is it hard to describe properties of algorithms? Procedia Computer Science. 2016. Vol. 101. P. 4–7. DOI: 10.1016/j.procs.2016.11.002.

4. MediaWiki. URL: `https://www.mediawiki.org` (accessed: 01.12.2022).

5. Wikipedia. URL: `https://wikipedia.org` (accessed: 01.12.2022).

6. Open Encyclopedia of Parallel Algorithmic Features. URL: `http://algowiki-project.org/en` (accessed: 01.12.2022).

7. Antonov A., Nikitenko D., Voevodin Vl. Algo500 — a New Approach to the Joint Analysis of Algorithms and Computers. Lobachevskii Journal of Mathematics. 2020. Vol. 41, no. 8. P. 1435–1443. DOI: 10.1134/S1995080220080041.

8. Antonov A.S., Maier R.V. Development and Implementation of the Algo500 Scalable Digital Platform Architecture. Lobachevskii J Math. 2022. Vol. 43. P. 837–847. DOI: 10.1134/S1995080222070058.

9. Voevodin V.V., Voevodin Vl.V. Parallel Computing. St. Petersburg, BHV-Petersburg, 2002. 608 p. (in Russian)

10. Antonov A.S., Volkov N.I. An AlgoView Web-visualization System for the AlgoWiki Project. Communications in Computer and Information Science. 2017. Vol. 753. P. 3–13. DOI: 10.1007/978-3-319-67035-5_1.

11. Antonov A., Volkov N. Interactive 3D Representation as a Method of Investigating Information Graph Features. Communications in Computer and Information Science. 2018. Vol. 965. P. 587–598. DOI: 10.1007/978-3-030-05807-4_50.

12. Antonov A.S., Volkov N.I. Information Graph Visualization Using AlgoView Software Tool. Lobachevskii J Math. 2020. Vol. 41, no. 6. P. 1427–1434. DOI: 10.1134/S199508022008003X.

13. Cholesky, A.-L. Sur la rèsolution numérique des systèmes déquations linéaires // La SABIX, Bulletins déja publiés, Sommaire du bulletin. 2005. No. 39. P. 81–95.

14. Antonov A., Frolov A., Konshin I., Voevodin Vl. Hierarchical Domain Representation in the AlgoWiki Encyclopedia: From Problems to Implementations. Communications in Computer and Information Science. 2018. Vol. 910. P. 3–15. DOI: 10.1007/978-3-319-99673-8_1.

15. Popov A., Nikitenko D., Antonov A., Voevodin Vl. Formal model of problems, methods, algorithms and implementations in the advancing AlgoWiki open encyclopedia. CEUR Workshop Proc. 2018. Vol. 2281. P. 1–11.

16. Antonov A.S., Maier R.V. A New Representation of Algorithmic Approaches in the AlgoWiki Encylopedia. Lobachevskii J Math. 2021. Vol. 42, no. 7. P. 1483–1491. DOI: 10.1134/S1995080221070039.

17. Antonov A., Voevodin Vad., Voevodin Vl., Teplov A. A Study of the Dynamic Characteristics of Software Implementation as an Essential Part for a Universal Description of Algorithm Properties. 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing Proceedings, February 17–19, 2016. P. 359–363. DOI: 10.1109/PDP.2016.24.

18. Voevodin V.V., Shvets P.A. Covering method for assessing the locality of data use in programs. Bulletin of the Ufa State Aviation Technical University. 2014. Vol. 18, no. 1(62). P. 224–229. (in Russian)

19. Antonov A., Teplov A. Generalized approach to scalability analysis of parallel applications. Lecture Notes in Computer Science. 2016. Vol. 10049. P. 291–304. DOI: 10.1007/978-3-319-49956-7_23.

20. Antonov A., Dongarra J., Voevodin V. AlgoWiki Project as an Extension of the Top500 Methodology. Supercomputing Frontiers and Innovations. 2018. Vol. 5, no. 1. P. 4–10. DOI: 10.14529/jsfi180101.

21. Home - | TOP500. URL: `https://top500.org` (accessed: 01.12.2022).

22. Dongarra J.J., Bunch J.R., Moler G.B., Stewart G.W. LINPACK Users' Guide. Society for Industrial and Applied Mathematics, 1979–1993.

23. Graph 500 | large-scale benchmarks. URL: `https://graph500.org` (accessed: 01.12.2022).

24. Murphy R.C., Wheeler K.B., Barrett B.W., Ang J.A. Introducing the Graph 500. Cray User's Group (CUG). May 5, 2010. Vol. 19. P. 45–74.

25. HPCG Benchmark. URL: `https://www.hpcg-benchmark.org` (accessed: 01.12.2022).

26. Heroux M., Dongarra J. Toward a New Metric for Ranking High Performance Computing Systems. UTK EECS Tech Report and Sandia National Labs Report SAND2013-4744, June 2013.

27. Voevodin V., Antonov A., Nikitenko D., *et al.* Supercomputer Lomonosov-2: Large Scale, Deep Monitoring and Fine Analytics for the User Community. Supercomputing Frontiers and Innovations. 2019. Vol. 6, no. 2. P. 4–11. DOI: 10.14529/jsfi190201.

СВЕДЕНИЯ ОБ ИЗДАНИИ

Научный журнал «Вестник ЮУрГУ. Серия «Вычислительная математика и информатика» основан в 2012 году.

Учредитель — Федеральное государственное автономное образовательное учреждение высшего образования «Южно-Уральский государственный университет» (национальный исследовательский университет).

Главный редактор — Л.Б. Соколинский.

Свидетельство о регистрации ПИ ФС77-57377 выдано 24 марта 2014 г. Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций.

Журнал включен в Реферативный журнал и Базы данных ВИНИТИ; индексируется в библиографической базе данных РИНЦ. Журнал размещен в открытом доступе на Всероссийском математическом портале MathNet. Сведения о журнале ежегодно публикуются в международной справочной системе по периодическим и продолжающимся изданиям «Ulrich's Periodicals Directory».

Решением Президиума Высшей аттестационной комиссии Министерства образования и науки Российской Федерации журнал включен в «Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук» по научным специальностям и соответствующим им отраслям науки: 2.3.5 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей (физико-математические науки), 05.13.17 – Теоретические основы информатики (физико-математические науки).

Подписной индекс научного журнала «Вестник ЮУрГУ», серия «Вычислительная математика и информатика»: 10244, каталог «Пресса России». Периодичность выхода — 4 выпуска в год.

Адрес редакции, издателя: 454080, г. Челябинск, проспект Ленина, 76, Издательский центр ЮУрГУ, каб. 32.

ПРАВИЛА ДЛЯ АВТОРОВ

1. Правила подготовки рукописей и пример оформления статей можно загрузить с сайта серии `http://vestnikvmi.susu.ru`. **Статьи, оформленные без соблюдения правил, к рассмотрению не принимаются.**
2. Адрес редакционной коллегии научного журнала «Вестник ЮУрГУ», серия «Вычислительная математика и информатика»:

   Россия 454080, г. Челябинск, пр. им. В.И. Ленина, 76, ЮУрГУ, кафедра СП,

   зам. главного редактора Цымблеру М.Л.
3. Адрес электронной почты редакции: `vestnikvmi@susu.ru`
4. **Плата с авторов за публикацию рукописей не взимается, и гонорары авторам не выплачиваются.**